United States
Department of
Agriculture

Economic
Research
Service

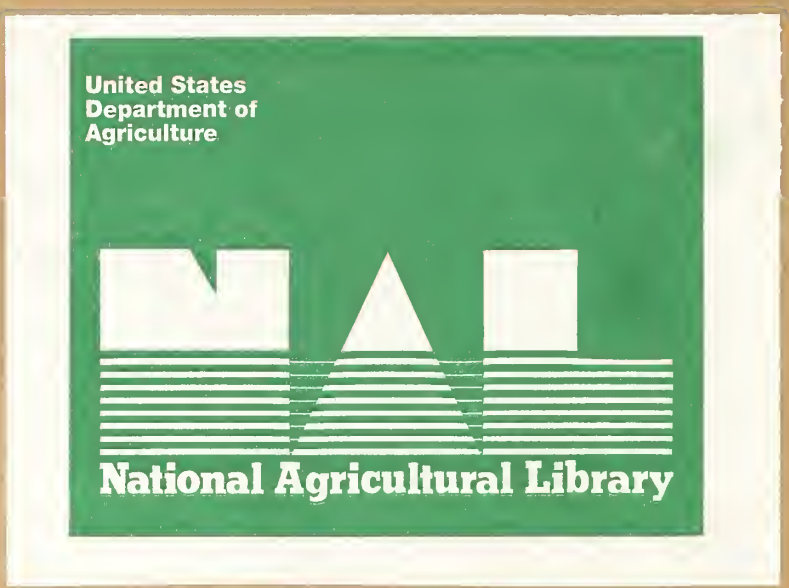
**International
Economics
Division**

IED Staff Report

Computer Programs to Support the World Grain, Oilseeds, and Livestock (GOL) Model

by

Vernon Roningen*
Karen Liu*
and
Francis Garvey*


United States
Department of
Agriculture
National Agricultural Library

Computer Programs to Support the World Grain, Oilseeds, and Livestock (GOL) Model

by

Vernon Roningen*
Karen Liu*
and
Francis Garvey*

March 1983

Policy Systems Section
Trade Policy Branch
International Economics Division
Economic Research Service
U.S. Department of Agriculture
Washington, D.C. 20250

*Chief of the Trade Policy Branch of the International Economics Division in ERS, Agricultural Economist, and former Computer Assistant, respectively in the Policy Systems Section of the Branch

Computer Programs to Support the World Grain, Oilseeds, and Livestock (GOL) Model by Vernon Roningen, Karen Liu, and Francis Garvey. International Economics Division, Economic Reserach Service, U.S. Department of Agriculture, Washington, D.C. 20250. Staff Report No. AGES830330, March 1983.

ABSTRACT

This report documents computer programs that can be used to create, update, and operate standard models designed to fit into ERS's World Grains, Oilseeds, and Livestock (GOL) model. The first part of the report outlines programs of TROLL commands which carry out tasks necessary to model development and operation. The second part of the report documents ways of creating sets of these TROLL commands using the mainframe or micro computer. To the extent that GOL component models have standard functional forms across commodities and countries, these groups of TROLL commands and parent programs greatly simplify GOL model management.

Table of Contents

Computer Programs to Support the World Grain, Oilseeds, and Livestock (GOL) Model

Introduction

The world Grain, Oilseeds, and Livestock (GOL) model is maintained in the TROLL computer simulation and econometric modeling package. This report illustrates the structured use of groups of TROLL commands to build and maintain standard GOL component models. Enough detail of typical support programs is given to aid the researcher in efficiently using TROLL for model building and operation, given his willingness to accept some standarization of model components. The report also presents alternative ways of creating groups of TROLL statements for various purposes on the mainframe and micro computer.

The structured TROLL commands and their parent programs simplify the creation, maintenance, and documentation of GOL model components. They are designed to make the full model/simulation and documentation capability of TROLL readily available to creators of GOL component models.

Assistance in the development and documentation of these programs was given by staff of the Trade Policy Branch including John Sullivan, Carol Stillwagon, William Kost, Rose Mayhew, and Theresa Wells, and John Gawalt.

I. Programs of TROLL Statements to Support a GOL Standard Country Model

I-A. General Description

Although TROLL is an interactive system, sets of computer programs containing TROLL commands can be submitted in the batch mode to carry out most computational tasks necessary for the construction and maintenance of standard GOL model components. 1/ The programs can be structured in a way to simplify the creation and management of standard country models. They allow for the cost-effective use of TROLL instructions for managing a large model in the batch mode.

This report contains illustrative TROLL programs used for the standard GOL model for the United States (USGOL). These programs may be cloned from the USGOL versions and text edited to any desired country nomenclature and to alternative purposes. Parts of the programs can be created on the micro computer. The documented programs are also useful guides to the use of TROLL commands to carry out various types of modeling tasks.

The TROLL programs have been set up to run as TROLL "MACRO" programs in order to eliminate unnecessary output and minimize costs. Other cost minimization features of TROLL, such as "in-core" calculations, have also been incorporated. Nevertheless, the size and complex nomenclature of a typical standard GOL model still means that many TROLL tasks may be expensive. Until experience is gained, most programs should be run in batch mode at the lowest priority. These types of programs can be submitted directly to batch through cards or can be put on TSO files and submitted to batch processing from the TSO interactive mode. By using the batch mode, considerable cost savings can be made. The programs have been named using TSO compatible naming conventions.

If structural changes (different equations, different variables, etc.) are made in the standard model, care has to be taken to insure that associated changes are also made in some of these support programs. TROLL user's manuals will have to be consulted regarding problems with individual TROLL commands.

1/ The reader should refer to appropriate TROLL manuals for details on TROLL commands. See: TROLL Reference Manual, Second Edition, Revised 1979, by the M.I.T. Center for Computational Research in Economics and Management Science, Cambridge, Massachusetts 02139. For details about operation of the IBM Time Sharing Option (TSO), consult the ERS Data Reference Center.

I-B. Documentation of TROLL Support Programs

The annotated program excerpts listed below are sets of TROLL commands which, when executed, perform various tasks needed for the creation or operation of a standard GOL country model. Complete documentation of TROLL commands can be found in the various TROLL user's guides and reference manuals. Comments above the program listing give a brief general description.

A "cloning" process for creating TROLL support programs or parts thereof, simply changes the country code, currency, and country name from U.S. nomenclature to the nomenclature for the selected country. Then a copy can be made for use with the selected country model.

In the presentation of a program excerpt, a short description is given followed by a listing. Long repetitive parts of the program listings have been omitted (marked by 3 vertical lines) to reduce the size of the documentation. However, enough of the listing is presented to give an indication of how various computational tasks are accomplished. The selected documentation serves as a guide to structured methods of managing a standard model as well as to the TROLL syntax that applies to the illustrated tasks. The documented TROLL statements which are illustrated are those applied to the USGOL version of the standard country model.

Typically, the programs begin with a TROLL OUTOPT command which sets printer options. Then comments which will be printed upon execution are sandwiched between a "PRINT" and "END" command. When these programs are submitted as TROLL "MACRO" programs, the printed message helps to identify the output. Program segments and the transition between segments are presented to highlight the syntax that must be observed.

A listing of the standard country model and its TROLL documentation can be found in a report by Roningen and Liu. 2/ Standard models presented in that report were largely created via the techniques presented in this report.

I-B.1. Programs to Create a Standard Model

A series of programs of TROLL statements can be used to create a standard country model and its supporting files. These statements are TROLL commands which enter model equations into TROLL using TROLL MODEDIT commands. Modification of the standard model for a country then can be done by inserting TROLL MODEDIT instructions into the programs CUSTOMIZ or by working interactively in TROLL.

2/ "The World Grain, Oilseeds, and Livestock (GOL) Model - Background and Standard Components," by Vernon Roningen and Karen Liu, ERS-IED Staff Report No. AGES830317, Washington, D.C. 1983.

I-B.1.a. USGOLESM - Enter a Standard Country Model into a TROLL File

This program enters a new country model into a TROLL file. It establishes the country symbol nomenclature and enters standard equations. It also serves as a backup file for a TROLL country model.

In order to change the standard model, it is better to use a systematic approach (as the illustrated in the program CUSTOMIZ) rather than tamper with the complicated syntax of this program on an ad hoc basis. If the country model is changed significantly from the standard model, the TROLL SOURCEMOD command can be used to prepare a new backup card deck of model creation instructions.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*************************************************************************
*
 USGOLESM - ENTER STANDARD GOL COUNTRY MODEL INTO A TROLL FILE.
*
*************************************************************************
&END
DELETE MODEL USGOL;
USEMOD USGOL;
MODEDIT ;
ADDSYM
ENDOGENOUS 'N 'NX 'NXS 'NPOL ,
DEFINITION 'DEF ,
EXOGENOUS 'X 'XN 'XNS 'XP ,
POLICY 'POL 'POLN 'POLP ,
FUNCTION 'F ,
COEFFICIENT 'C ,
PARAMETER 'P 'PX 'PPOL ,;
ADDEQ BOTTOM
USMDBF :   USMDBF'N = USMDBFI'C*(USPNG'X/USPDBF'N)**USMDBFPC'C*(USPNG'X(-1)/
USPDBF'N(-1))**USMDBFPL'C*USPDBF'N,
USMDPK :   USMDPK'N = USMDPKI'C*(USPNG'X/USPDPK'N)**USMDPKPC'C*(USPNG'X(-1)/
USPDPK'N(-1))**USMDPKPL'C*USPDPK'N,
USMDML :   USMDML'N = USMDMLI'C*(USPNG'X/USPDML'N)**USMDMLPC'C*(USPNG'X(-1)/
USPDML'N(-1))**USMDMLPL'C*USPDML'N,
USMDDM :   USMDDM'N = USMDDMI'C*(USPNG'X/USPDDM'N)**USMDDMPC'C*(USPNG'X(-1)/
USPDDM'N(-1))**USMDDMPL'C*USPDDM'N,
USMDPM :   USMDPM'N = USMDPMI'C*(USPNG'X/USPDPM'N)**USMDPMPC'C*(USPNG'X(-1)/
USPDPM'N(-1))**USMDPMPL'C*USPDPM'N,
```

← Establish printout characteristics

Print title to mark program

Delete old model

Establish new model name

Enter the TROLL MODEDIT mode

Add symbols for the model

Add equations to the model

```
USPDSB :   USPDSB'N = IF USPESB'DEF LT USPTSB'XNS+USMTSB'N+USTMSB'POLN+
USTCSB'POLN AND USPESB'DEF GT USPTSB'XNS-USMTSB'N+USMDSB'N-USTESB'POLN+
USTCSB'POLN OR USQTSB'N GE USEQSB'POLN OR USQTSB'N LE -USMQSB'POLN THEN
(((ABSV'F(USPESB'DEF)+USPESB'DEF)/2)+0.01)
            ELSE (IF USQTSB'N LT 0 THEN ABSV'F(USPTSB'XNS+USMTSB'N+USTMSB'POLN+
USTCSB'POLN) ELSE ABSV'F(USPTSB'XNS-USMTSB'N+USMDSB'N-USTESB'POLN+USTCSB'POLN)),
USPDOS :   USPDOS'N = IF USPEOS'DEF LT USPTOS'XNS+USMTOS'N+USTMOS'POLN+
USTCOS'POLN AND USPEOS'DEF GT USPTOS'XNS-USMTOS'N+USMDOS'N-USTEOS'POLN+
USTCOS'POLN OR USQTOS'N GE USEQOS'POLN OR USQTOS'N LE -USMQOS'POLN THEN
(((ABSV'F(USPEOS'DEF)+USPEOS'DEF)/2)+0.01)
            ELSE (IF USQTOS'N LT 0 THEN ABSV'F(USPTOS'XNS+USMTOS'N+USTMOS'POLN+
USTCOS'POLN) ELSE ABSV'F(USPTOS'XNS-USMTOS'N+USMDOS'N-USTEOS'POLN+USTCOS'POLN)),
USPDSM :   USPDSM'N = IF USPESM'DEF LT USPTSM'XNS+USMTSM'N+USTMSM'POLN+
USTCSM'POLN AND USPESM'DEF GT USPTSM'XNS-USMTSM'N+USMDSM'N-USTESM'POLN+
USTCSM'POLN OR USQTSM'N GE USEQSM'POLN OR USQTSM'N LE -USMQSM'POLN THEN
(((ABSV'F(USPESM'DEF)+USPESM'DEF)/2)+0.01)
            ELSE (IF USQTSM'N LT 0 THEN ABSV'F(USPTSM'XNS+USMTSM'N+USTMSM'POLN+
USTCSM'POLN) ELSE ABSV'F(USPTSM'XNS-USMTSM'N+USMDSM'N-USTESM'POLN+USTCSM'POLN)),
USPDSO :   USPDSO'N = IF USPESO'DEF LT USPTSO'XNS+USMTSO'N+USPTSO'XNS*
USTMSO'POLN/100+USTCSO'POLN AND USPESO'DEF GT USPTSO'XNS-USMTSO'N+USMDSO'N-
USTESO'POLN+USTCSO'POLN OR USQTSO'N GE USEQSO'POLN OR USQTSO'N LE -USMQSO'POLN
 THEN (((ABSV'F(USPESO'DEF)+USPESO'DEF)/2)+0.01)
                  ELSE (IF USQTSO'N LT 0 THEN
 ABSV'F(  USPTSO'XNS+USMTSO'N+USPTSO'XNS*
USTMSO'POLN/100+USTCSO'POLN) ELSE
 ABSV'F(USPTSO'XNS-USMTSO'N+USMDSO'N-USTESO'POLN+
USTCSO'POLN)),
USPDOM :   USPDOM'N = IF USPEOM'DEF LT USPTOM'XNS+USMTOM'N+USTMOM'POLN+
USTCOM'POLN AND USPEOM'DEF GT USPTOM'XNS-USMTOM'N+USMDOM'N-USTEOM'POLN+
USTCOM'POLN OR USQTOM'N GE USEQOM'POLN OR USQTOM'N LE -USMQOM'POLN THEN
(((ABSV'F(USPEOM'DEF)+USPEOM'DEF)/2)+0.01)
            ELSE (IF USQTOM'N LT 0 THEN ABSV'F(USPTOM'XNS+USMTOM'N+USTMOM'POLN+
USTCOM'POLN) ELSE ABSV'F(USPTOM'XNS-USMTOM'N+USMDOM'N-USTEOM'POLN+USTCOM'POLN)),
USPDOO :   USPDOO'N = IF USPEOO'DEF LT USPTOO'XNS+USMTOO'N+USTMOO'POLN+
USTCOO'POLN AND USPEOO'DEF GT USPTOO'XNS-USMTOO'N+USMDOO'N-USTEOO'POLN+
USTCOO'POLN OR USQTOO'N GE USEQOO'POLN OR USQTOO'N LE -USMQOO'POLN THEN
(((ABSV'F(USPEOO'DEF)+USPEOO'DEF)/2)+0.01)
            ELSE (IF USQTOO'N LT 0 THEN ABSV'F(USPTOO'XNS+USMTOO'N+USTMOO'POLN+
USTCOO'POLN) ELSE ABSV'F(USPTOO'XNS-USMTOO'N+USMDOO'N-USTEOO'POLN+USTCOO'POLN)),
USPDDB :   USPDDB'N = IF USPEDB'DEF LT USPTDB'XNS+USMTDB'N+(IF -USQTDB'N LT
USTQDB'PPOL THEN USTMDB'POLN ELSE USTQDBM'PPOL*USTMDB'POLN)+USTCDB'POLN AND
USPEDB'DEF GT USPTDB'XNS-USMTDB'N+USMDDB'N-USTEDB'POLN+USTCDB'POLN OR USQTDB'N
 GE USEQDB'POLN OR USQTDB'N LE -USMQDB'POLN THEN
(((ABSV'F(USPEDB'DEF)+USPEDB'DEF)/2)+0.01) ELSE (IF USQTDB'N
 LT 0 THEN ABSV'F(
 USPTDB'XNS+USMTDB'N+(IF -USQTDB'N LT USTQDB'PPOL THEN USTMDB'POLN
 ELSE USTQDBM'PPOL*USTMDB'POLN)+USTCDB'POLN) ELSE
 ABSV'F(USPTDB'XNS-USMTDB'N+USMDDB'N-
USTEDB'POLN+USTCDB'POLN)),
USPDDC :   USPDDC'N = IF USPEDC'DEF LT USPTDC'XNS+USMTDC'N+USPTDC'XNS*
USTMDC'POLN/100+USTCDC'POLN AND USPEDC'DEF GT USPTDC'XNS-USMTDC'N+USMDDC'N-
USTEDC'POLN+USTCDC'POLN OR USQTDC'N GE USEQDC'POLN OR USQTDC'N LE -USMQDC'POLN
 THEN (((ABSV'F(USPEDC'DEF)+USPEDC'DEF)/2)+0.01)
                  ELSE (IF USQTDC'N LT 0 THEN
      ABSV'F(                     USPTDC'XNS+USMTDC'N+USPTDC'XNS*
USTMDC'POLN/100+USTCDC'POLN) ELSE
 ABSV'F(USPTDC'XNS-USMTDC'N+USMDDC'N-USTEDC'POLN+
USTCDC'POLN)),
USPDDD :   USPDDD'N = IF USPEDD'DEF LT USPTDD'XNS+USMTDD'N+USTMDD'POLN+
USTCDD'POLN AND USPEDD'DEF GT USPTDD'XNS-USMTDD'N+USMDDD'N-USTEDD'POLN+
USTCDD'POLN OR USQTDD'N GE USEQDD'POLN OR USQTDD'N LE -USMQDD'POLN THEN
(((ABSV'F(USPEDD'DEF)+USPEDD'DEF)/2)+0.01)
             ELSE (IF USQTDD'N LT 0 THEN
    ABSV'F(                     USPTDD'XNS+USMTDD'N+USTMDD'POLN+
USTCDD'POLN) ELSE
 ABSV'F(USPTDD'XNS-USMTDD'N+USMDDD'N-USTEDD'POLN+USTCDD'POLN)),
 ;
PRINT ALL COMMENT;
FILEMOD USGOL;
```

End the ADDEQ routine

Print the model

File the model permanently

I-B.1.b. USGOLEQC - Enter Equation Comments into the Standard Model

Equation comments can be added to the standard model by this program of TROLL MODEDIT commands. These comments are not necessary for model operation but they label each equation so that it is much easier to read a listing of the model.

The comments are placed at the equation labelled (in the model) by the symbolic name appearing after the equation number. This symbol is given directly after the EQCOM command. These statements assume that equation symbolic names have been entered with the equations in USGOLESM. The MODCOM statement at the beginning of the program adds the general GOL notation to the standard model as a model comment. If equation comments are lost because of model edits, they can be restored again by re-running this program.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*************************************************************************
*
 USGOLEQC-ENTER EQUATION COMMENTS
*
&END
USEMOD USGOL;
MODEDIT;
MODCOM   | <------Enter the model comment
-------------------------------------------------------------------------
| NOTATION FOR ERS GRAIN, OILSEED, AND LIVESTOCK (GOL) MODEL :  SYMBOL AND  |
| VARIABLE NAMES CONTAIN UP TO 8 CHARACTERS AND ARE FOLLOWED BY A SUFFIX    |
| WHICH SHOWS THE DECLARATION (E.G. CONSTANT, EXOGENOUS VARIABLE, ETC.).  THE |
| FIRST 2 CHARACTERS ARE THE COUNTRY CODE AND THE NEXT 2, AN EQUATION 'TYPE'  |
| CODE, THE NEXT 2 CHARACTERS ARE USUALLY A 2 DIGIT COMMODITY CODE.  AN      |
| ELASTICITY WILL HAVE 2 MORE CHARACTERS INDICATING THE CODE TO WHICH THE    |
 USGOLEQC - ENTER EQUATION COMMENTS FOR A STANDARD COUNTRY MODEL
| ELASTICITY RELATES.  GENERALLY, THE NUMBER OF CHARACTERS IN A SYMBOL HAS   |
| A MEANING:   5 CHAR, - COUNTRY SPECIFIC VARIABLE, 6 CHAR, - COUNTRY AND    |
| COMMODITY SPECIFIC VARIABLE, 7 CHAR,(ENDING WITH 'I') - EQUATION INTERCEPT, |
| 8 CHAR, - COEFFICIENT/ELASTICITY.  COMMODITY CODES ARE:  BF - BEEF+VEAL,   |
| PK - PORK, ML - MUTTON+LAMB(+GOAT), DM - DAIRY-MILK, PM - POULTRY-MEAT,    |
| PE - POULTRY-EGGS, WH- WHEAT, CN - CORN, CG - OTHER COARSE GRAINS,         |
| RI - RICE, SB- SOYBEANS, OS - OTHER OILSEEDS, SM - SOYMEAL, SO - SOYOIL,   |
| OM - OTHER MEALS, OO - OTHER OILS, DB - DAIRY-BUTTER, DC - DAIRY-CHEESE,   |
| DO - DAIRY-OTHER PRODUCTS.  EQUATION 'TYPE' CODES ARE:  MD - MARGIN-DOMESTIC, |
| MT - MARGIN-TRADE, PS - PRICE-SUPPLY, AR - AREA, YD - YIELD,               |
| QS - QUANTITY-SUPPLIED, QC - QUANTITY-CRUSHED, FC - FEED COST,             |
| LN - LIVESTOCK-NUMBERS, LA - LIVESTOCK-ADDITIONS, LS - LIVESTOCK-SLAUGHTER, |
| QF - QUANTITY-FED, QD - QUANTITY-FOOD AND OTHER DEMAND, SK - ENDING STOCKS, |
| QT - QUANTITY-TRADED, PE - PRICE ESTIMATE (DEMAND) WITH TRADE RESTRICTIONS, |
| PD - PRICE-DEMAND.  POLICY VARIABLE CODES ARE:  EQ - EXPORT QUOTA,         |
| MQ - IMPORT QUOTA, TE - TAX-EXPORTS, TM - TAX-IMPORTS, TP - TAX-PRODUCTION, |
| TC - TAX-CONSUMPTION. |-------------------------------------------------
-----------------------;
```

```
EQCOM USMDBF *
*****************************************************************************
*                                                                           *
*  GRAIN, OILSEEDS, LIVESTOCK COUNTRY MODEL - UNITED STATES (US)            *
*                                                                           *
*
*
*
*****************************************************************************
*                                                                           *
*  DOMESTIC MARGIN EQUATIONS                                                *
*                                                                           *
*****************************************************************************
*                    *                                                      *
MARGIN (DOMESTIC) = BEEF+VEAL;
EQCOM USMDPK MARGIN (DOMESTIC) = PORK;
EQCOM USMDML MARGIN (DOMESTIC) = MUTTON+LAMB;

|                                  |                                  |


EQCOM USPEDC PRICE ESTIMATE (DEMAND) WITH RESTRICTED TRADE - DAIRY=CHEESE;
EQCOM USPEDO PRICE ESTIMATE (DEMAND) WITH RESTRICTED TRADE - DAIRY=OTHER PROD.;
EQCOM USPDBF *
*****************************************************************************
*                                                                           *
*  DEMAND PRICE EQUATIONS - TRADE LINKED OR DOMESTIC MARKET CLEARING ESTIMATES *
*                                                                           *
*****************************************************************************
*                    *                                                      *
PRICE (DEMAND) = BEEF+VEAL;
EQCOM USPDPK PRICE (DEMAND) = PORK;
EQCOM USPDML PRICE (DEMAND) = MUTTON+LAMB;
EQCOM USPDDM PRICE (DEMAND) = DAIRY=MILK;
EQCOM USPDPM PRICE (DEMAND) = POULTRY=MEAT;
EQCOM USPDPE PRICE (DEMAND) = POULTRY=EGGS;
EQCOM USPDWH PRICE (DEMAND) = WHEAT;
EQCOM USPDCN PRICE (DEMAND) = CORN;
EQCOM USPDCG PRICE (DEMAND) = OTHER COARSE GRAINS;
EQCOM USPDRI PRICE (DEMAND) = RICE;
EQCOM USPDSB PRICE (DEMAND) = SOYBEANS;
EQCOM USPDOS PRICE (DEMAND) = OTHER OILSEEDS;
EQCOM USPDSM PRICE (DEMAND) = SOYMEAL;
EQCOM USPDSO PRICE (DEMAND) = SOYOIL;
EQCOM USPDOM PRICE (DEMAND) = OTHER MEALS;
EQCOM USPDOO PRICE (DEMAND) = OTHER OILS;
EQCOM USPDDB PRICE (DEMAND) = DAIRY=BUTTER;
EQCOM USPDDC PRICE (DEMAND) = DAIRY=CHEESE;
EQCOM USPDDO PRICE (DEMAND) = DAIRY=OTHER PRODUCTS;
EQCOM USPTBF *
*****************************************************************************
*                                                                           *
*  TRADE PRICE LINKAGE EQUATIONS                                            *
*                                                                           *
*****************************************************************************
*                    *
TRADE PRICE LINKAGE = BEEF+VEAL;
EQCOM USPTPK TRADE PRICE LINKAGE = PORK ;
EQCOM USPTML TRADE PRICE LINKAGE = MUTTON+LAMB ;
EQCOM USPTPM TRADE PRICE LINKAGE = POULTRY=MEAT ;
EQCOM USPTPE TRADE PRICE LINKAGE = POULTRY=EGGS ;
EQCOM USPTWH TRADE PRICE LINKAGE = WHEAT ;
EQCOM USPTCN TRADE PRICE LINKAGE = CORN ;
EQCOM USPTCG TRADE PRICE LINKAGE = OTHER COARSE GRAINS ;
EQCOM USPTRI TRADE PRICE LINKAGE = RICE ;
EQCOM USPTSB TRADE PRICE LINKAGE = SOYBEANS ;
EQCOM USPTOS TRADE PRICE LINKAGE = OTHER OILSEEDS ;
EQCOM USPTSM TRADE PRICE LINKAGE = SOYMEAL ;
EQCOM USPTSO TRADE PRICE LINKAGE = SOYOIL ;
EQCOM USPTOM TRADE PRICE LINKAGE = OTHER MEALS ;
EQCOM USPTOO TRADE PRICE LINKAGE = OTHER OILS ;
EQCOM USPTDB TRADE PRICE LINKAGE = DAIRY=BUTTER ;
EQCOM USPTDC TRADE PRICE LINKAGE = DAIRY=CHEESE ;
EQCOM USPTDO TRADE PRICE LINKAGE = DAIRY=OTHER PRODUCTS ;
FILEMOD USGOL;
```

Enter equation comments

I-B.1.c. USGOLSMC - Enter Symbol Comments into the Standard Model

Comments for documentation purposes are entered for each variable and constant symbol in the standard model by this program.

The "comment" or symbol declaration list attached to a model by this program serves as a dictionary for all symbols used in the model.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*****************************************************************************
*
 USGOLSMC - ENTER SYMBOL COMMENTS FOR A STANDARD COUNTRY MODEL
*
*****************************************************************************
&END
USEMOD USGOL;
MODEDIT ;
SYMCOM USMDBF MARGIN (DOMESTIC) * BEEF+VEAL (US$/MT);
SYMCOM USMDPK MARGIN (DOMESTIC) * PORK (US$/MT);
```

Enter symbol comments

```
SYMCOM USTMDO TARIFF(+)/SUBSIDY(-) (IMPORT) * DAIRY-OTHER PRODUCTS (US$/MT);
SYMCOM USTEBF TAX(+)/SUBSIDY(-) (EXPORT) * BEEF+VEAL (US$/MT);
SYMCOM USTEPK TAX(+)/SUBSIDY(-) (EXPORT) * PORK (US$/MT);
SYMCOM USTEML TAX(+)/SUBSIDY(-) (EXPORT) * MUTTON+LAMB (US$/MT);
SYMCOM USTEPM TAX(+)/SUBSIDY(-) (EXPORT) * POULTRY-MEAT (US$/MT);
SYMCOM USTEPE TAX(+)/SUBSIDY(-) (EXPORT) * POULTRY-EGGS (US$/MT);
SYMCOM USTEWH TAX(+)/SUBSIDY(-) (EXPORT) * WHEAT (US$/MT);
SYMCOM USTECN TAX(+)/SUBSIDY(-) (EXPORT) * CORN (US$/MT);
SYMCOM USTECG TAX(+)/SUBSIDY(-) (EXPORT) * OTHER COARSE GRAINS (US$/MT);
SYMCOM USTERI TAX(+)/SUBSIDY(-) (EXPORT) * RICE (US$/MT);
SYMCOM USTESB TAX(+)/SUBSIDY(-) (EXPORT) * SOYBEANS (US$/MT);
SYMCOM USTEOS TAX(+)/SUBSIDY(-) (EXPORT) * OTHER OILSEEDS (US$/MT);
SYMCOM USTESM TAX(+)/SUBSIDY(-) (EXPORT) * SOYMEAL (US$/MT);
SYMCOM USTESO TAX(+)/SUBSIDY(-) (EXPORT) * SOYOIL (US$/MT);
SYMCOM USTEOM TAX(+)/SUBSIDY(-) (EXPORT) * OTHER MEALS (US$/MT);
SYMCOM USTEOO TAX(+)/SUBSIDY(-) (EXPORT) * OTHER OILS (US$/MT);
SYMCOM USTEDB TAX(+)/SUBSIDY(-) (EXPORT) * DAIRY-BUTTER (US$/MT);
SYMCOM USTEDC TAX(+)/SUBSIDY(-) (EXPORT) * DAIRY-CHEESE (US$/MT);
SYMCOM USTEDO TAX(+)/SUBSIDY(-) (EXPORT) * DAIRY-OTHER PRODUCTS (US$/MT);
SYMCOM USTODB  TARIFF-QUOTA * DAIRY-BUTTER ;
SYMCOM USTODBM TARIFF-QUOTA PARAMETER * DAIRY-BUTTER;
FILEMOD USGOL;
```

I-B.1.d. USGOLCEM - Create Elasticity Matrices for the Documentation of the Standard Model and Enter Coefficients

This program of TROLL MATRIX and LEDIT commands can be run to create elasticity matrices which help to document the standard country model. The standard model can be understood by reading the documented TROLL equations in a listing of the model. However, since equations are standardized, the equation structure in terms of elasticities and coefficients can be summarized in matrix form. There is one matrix for the food demand system, one for feed demand, etc. Programs which enter the elasticities also update these matrices. Whenever these matrices are printed out, they will give all of the current values of elasticities and constants embedded in the model. The matrix table for elasticities is also useful because some of the properties of the model (e.g. the existence of elasticity symmetry) are embedded in the elasticities themselves rather than in the equation forms.

If constant values are changed by means other than these programs, care should be taken that the matrices displaying the constants are also updated. Otherwise, it is easy to lose track of the coefficients actually used in the model equations.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*************************************************************************
*
USGOLCEM - CREATE MATRICES TO DISPLAY CONSTANTS AND PARAMETERS FOR THE
STANDARD COUNTRY MODEL- USGOL.
CREATE ZERO VALUED CONSTANTS FOR THE STANDARD COUNTRY MODEL.
THE MATRICES CREATED ARE:
USMAELAS - MARGIN ELASTICITIES
USARELAS - CROP AREA ELASTICITIES
USYDELAS - CROP YIELD ELASTICITIES
USOLELAS - OILSEED PARAMETERS AND ELASTICITIES
USFCELAS - FEED COST PARAMETERS
USLPELAS - LIVESTOCK AND LIVESTOCK PRODUCT ELASTICITIES
USDPELAS - DAIRY PRODUCT ELASTICITIES
USFPELAS - FEED DEMAND PARAMETERS
USOFELAS - FEED DEMAND ELASTICITIES
USOIELAS - INDUSTRIAL DEMAND ELASTICITIES
USODELAS - FOOD AND NON-FEED ELASTICITIES
USSKELAS - STOCK ELASTICITIES
USPEELAS - PRICE ESTIMATE DAMPING PARAMETERS
USPDELAS - DEMAND PRICE ELASTICITIES
*
*************************************************************************
&END
```

```
DELETE DATA USMAELAS ;
DO USMAELAS = CRMAT( 19 , 6  ,NA);
DELETE GENERAL LABEL_USMACOLE;
LEDIT USMACOLE ;
ADD TOP, INTERCEPTS FOR DOMESTIC MARGIN EQUATIONS (I),
 DOMESTIC MARGIN ELASTICITIES WRT CURRENT NON-GOL PRICES (PC),
 DOMESTIC MARGIN ELASTICITIES WRT LAGGED NON-GOL PRICES (PL),
 INTERCEPTS FOR TRADE MARGIN EQUATIONS (I),
 TRADE MARGIN ELASTICITIES WRT CURRENT NON-GOL PRICES (PC),
 TRADE MARGIN ELASTICITIES WRT LAGGED NON-GOL PRICES (PL);
FILE ;
DELETE GENERAL LABEL_USMAROWE;
LEDIT USMAROWE ;
ADD TOP,
 1 BEEF+VEAL (BF) ---------- ,
 2 PORK (PK) ---------- ,
 3 MUTTON+LAMB (ML) ---------- ,
 4 DAIRY-MILK (DM) ---------- ,
 5 POULTRY-MEAT (PM) ---------- ,
 6 POULTRY-EGGS (PE) ---------- ,
 7 WHEAT (WH) ---------- ,
 8 CORN (CN) ---------- ,
 9 OTHER COARSE GRAINS (CG) ---- ,
 10 RICE (RI) ---------- ,
 11 SOYBEANS (SB) ---------- ,
 12 OTHER OILSEEDS (OS) ---------- ,
 13 SOYMEAL (SM) ---------- ,
 14 SOYOIL (SO) ---------- ,
 15 OTHER MEALS (OM) ---------- ,
 16 OTHER OILS (OO) ---------- ,
 17 DAIRY-BUTTER (DB) ---------- ,
 18 DAIRY-CHEESE (DC) ---------- ,
 19 DAIRY-OTHER PRODUCTS (DO) ;
FILE ;
DEDIT USMAELAS;
COMMENT USMAELAS = MATRIX OF MARGIN SHARE ELASTICITIES - UNITED STATES ;
FILE ;
DELETE GENERAL LABEL_USMATITL;
LEDIT USMATITL ;
ADD TOP, DOMESTIC AND TRADE MARGIN ELASTICITIES - UNITED STATES /1,
1/ MARGIN SHARE OF DEMAND PRICE WRT TO (NON-GOL/DEMAND) PRICE;
FILE ;
DO OPRTMAT(USMAELAS,USMAROWE'L,USMACOLE'L,0,0,-28 ,USMATITL'L,1);
DELETE DATA USARELAS ;
DO USARELAS = CRMAT( 7 ,10, NA);
DELETE GENERAL LABEL_USARCOLE ;
```

Create a matrix named USMAELAS

Create labels for the columns of the matrix

Create labels for the rows of the matrix

Enter a comment to mark the matrix data file

Create a title and footnote for the matrix

Print out the matrix USMAELAS

```
DEDIT USPDELAS;
COMMENT USPDELAS = MATRIX OF INTERCEPT TERMS FOR DEMAND PRICE
 EQUATIONS - UNITED STATES;
FILE ;
DELETE GENERAL LABEL_USPDTITL;
LEDIT USPDTITL ;
ADD TOP, INTERCEPTS FOR DEMAND PRICE EQUATIONS - UNITED STATES;
FILE;
DO OPRTMAT(USPDELAS,USDDROWE'L,USPDCOLE'L,0,0,-28,USPDTITL'L,1);
DELETE CONST   GOL;
CEDIT   GOL;
COMMENT ELASTICITIES, PARAMETERS, INTERCEPTS, AND OTHER CONSTANTS FOR
GOL MODEL;
ADD
 USARCGCG  0.     USARCGCN  0.     USARCGI   0.
 USARCGDS  0.     USARCGRI  0.     USARCGSB  0.
 USARCGWH  0.     USARCNCG  0.     USARCNCN  0.
 USARCNI   0.     USARCNDS  0.     USARCNRI  0.

 USYDSBAR  0.     USYDSBI   0.     USYDSBSB  0.
 USYDSBTR  0.     USYDWHAR  0.     USYDWHI   0.
 USYDWHTR  0.     USYDWHWH  0;
PRINT ALL;
FILE ;
```

Create a constant file

Add a comment to the file

Add constants to the file

Print the constant file

File the constants permanently

A sample display matrix created by the above program is presented below. Other programs will replace the NA's with elasticity/coefficient values.

DOMESTIC AND TRADE MARGIN ELASTICITIES - UNITED STATES /1

	INTERCEPTS FOR DOMESTIC MARGIN EQUATIONS (I)	DOMESTIC MARGIN ELASTICITIES WRT CURRENT NON-GOL PRICES (PC)	DOMESTIC MARGIN ELASTICITIES WRT LAGGED NON-GOL PRICES (PL)	INTERCEPTS FOR TRADE MARGIN EQUATIONS (I)	TRADE MARGIN ELASTICITIES WRT CURRENT NON-GOL PRICES (PC)	TRADE MARGIN ELASTICITIES WRT LAGGED NON-GOL PRICES (PL)
1 BEEF+VEAL (BP)	NA	NA	NA	NA	NA	NA
2 PORK (PK)	NA	NA	NA	NA	NA	NA
3 MUTTON+LAMB (ML)	NA	NA	NA	NA	NA	NA
4 DAIRY-MILK (DM)	NA	NA	NA	NA	NA	NA
5 POULTRY-MEAT (PM)	NA	NA	NA	NA	NA	NA
6 POULTRY-EGGS (PE)	NA	NA	NA	NA	NA	NA
7 WHEAT (WH)	NA	NA	NA	NA	NA	NA
8 CORN (CN)	NA	NA	NA	NA	NA	NA
9 OTHER COARSE GRAINS (CG)	NA	NA	NA	NA	NA	NA
10 RICE (RI)	NA	NA	NA	NA	NA	NA
11 SOYBEANS (SB)	NA	NA	NA	NA	NA	NA
12 OTHER OILSEEDS (OS)	NA	NA	NA	NA	NA	NA
13 SOYMEAL (SM)	NA	NA	NA	NA	NA	NA
14 SOYOIL (SO)	NA	NA	NA	NA	NA	NA
15 OTHER MEALS (OM)	NA	NA	NA	NA	NA	NA
16 OTHER OILS (OO)	NA	NA	NA	NA	NA	NA
17 DAIRY-BUTTER (DB)	NA	NA	NA	NA	NA	NA
18 DAIRY-CHEESE (DC)	NA	NA	NA	NA	NA	NA
19 DAIRY-OTHER PRODUCTS (DO)	NA	NA	NA	NA	NA	NA

1/ MARGIN SHARE OF DEMAND PRICE WRT TO (NON-GOL/DEMAND) PRICE

I-B.1.e. CUSTOMIZ - Customize a Standard Model to Fit a Particular Agricultural Economy

This program is an example of a tool which can be used to modify a standard country model. The TROLL commands in the sample listing are those which were used to customize the standard model to fit some particular aspects of the U.S. agricultural economy. Changes in the standard model are made via TROLL MODEDIT commands.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*************************************************************************
*
CUSTOMIZ  - CUSTOMIZE A STANDARD MODEL TO PARTICULAR COUNTRY CONDITIONS.
*
*************************************************************************
&END
USEMOD USGOL;
MODEDIT;
CHANGEQ $USTMSO'POLN$(USPTSO'XNS*USTMSO'POLN/100)$ G USPDSO;
CHANGEQ $USTMDC'POLN$(USPTDC'XNS*USTMDC'POLN/100)$ G USPDDC;
CHANGEQ $USTMDB'POLN$(IF -USOTDB'N   LT USTODB'PPOL THEN USTMDB'POLN
 ELSE USTODBM'PPOL*USTMDB'POLN)$ G USPDDB;
DELEQ 153 TO 155;
ADDEQ 152,
USSKDB :   USSKDBI'C*(USPDDB'N/USPNG'X)**USSKDBDB'C*(USPRDM'POLN/
USPSDM'DEF)**USSKDBPR'C*(USODDB'N+USOSDB'N),
USSKDC :   USSKDCI'C*(USPDDC'N/USPNG'X)**USSKDCDC'C*(USPRDM'POLN/
USPSDM'DEF)**USSKDCPR'C*(USODDC'N+USOSDC'N),
USSKDO :   USSKDOI'C*(USPDDO'N/USPNG'X)**USSKDODO'C*(USPRDM'POLN/
USPSDM'DEF)**USSKDOPR'C*(USODDO'N+USOSDO'N);
SYMCOM USPRDM PARITY RATIO * DAIRY-MILK (%);
SYMCOM USTODB TARIFF-QUOTA * DAIRY-BUTTER;
SYMCOM USTODBM TARIFF-QUOTA MULTIPLIER * DAIRY-BUTTER;
SYMCOM USSKDBPR STOCK ELASTICITY * DAIRY-BUTTER WRT MILK PARITY RATIO;
SYMCOM USSKDCPR STOCK ELASTICITY * DAIRY-CHEESE WRT MILK PARITY RATIO;
SYMCOM USSKDOPR STOCK ELASTICITY * DAIRY-OTHER PRODUCTS WRT MILK PARITY RATIO;
FILEMOD USGOL;
```

Select model to be used

Change equations in the model

Delete equations from the model

Add new equations to the model after equation 152

Add symbol comments for the new equations

File the changed model permanently

I-B.2. Programs for Entering Data into a Country TROLL File

A country model requires a data file for each endogenous and exogenous variable. These programs can be used to enter data into the country TROLL file in a systematic way. They contain standard TROLL DEDIT commands which enter data and comments into data files.

The data entry programs are grouped by type of data for convenience. Endogenous data is needed at minimum for two years and more if the model is to be validated. Exogenous and policy data are needed for the length of projection desired. Data can also be entered interactively but the cost will be much higher than if these programs are used in the batch mode. Note that each program begins with a general TROLL command to delete all previous data for the group prior to the entry of new data.

I-B.2.a. USGOLEND - Enter Endogenous Data

A standard country model requires two types of balance data. First, quantity supply and utilization data is needed; second, supply-demand price balances are needed. Spread-sheet programs on a microcomputer can help assemble balanced data sets from a minimum amount of input data. Then programs such as USGOLEND can be used to insert the minimum amount of this "balanced" quantity and price data into a country TROLL file.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 !
&PRINT
*************************************************************************
*
 USGOLEND - ENTER DATA FOR ENDOGENOUS VARIABLES
*
*************************************************************************
&END
DELETE DATA USMT**!     <-- Delete all 6 letter data files beginning with USMT
DEDIT USMTBF  ,1,1975!
COMMENT  MARGIN (TRADE) * BEEF+VEAL (US$/MT)                              !
DATA     2285.000    1894.000    1942.000    2236.000    2456.000    2745.000 !
FILE!
DEDIT USMTPK  ,1,1975!
COMMENT  MARGIN (TRADE) * PORK (US$/MT)                                   !
DATA     1223.000    1289.000    1382.000    1559.000    1594.000    1535.000 !
FILE!
DEDIT USMTML  ,1,1975!
COMMENT  MARGIN (TRADE) * MUTTON+LAMB (US$/MT)                            !
DATA     2338.000    2718.000    2455.000    3061.000    3315.000    3159.000 !
FILE!
DEDIT USMTPM  ,1,1975!
COMMENT  MARGIN (TRADE) * POULTRY-MEAT (US$/MT)                           !
DATA      413.000     466.000     456.000     500.000     566.000     554.000 !
FILE!
```

Create data files, add comments, add data, and file permanently

```
DEDIT USPDDC  ,1,1975!
COMMENT  PRICE (DEMAND) * DAIRY+CHEESE (US$/MT)                           !
DATA     3382.000    3788.000    3792.000    4056.000    4586.000    5058.000 !
FILE!
DEDIT USPDDO  ,1,1975!
COMMENT  PRICE (DEMAND) * DAIRY-OTHER PRODUCTS (US$/MT)                   !
DATA     2908.000    2796.000    2984.000    3151.000    3469.000    3469.000 !
FILE!
OPRTDATA ALPHA  USMT**!
OPRTDATA ALPHA  USMD**!
OPRTDATA ALPHA  USAR**!
OPRTDATA ALPHA  USYD**!
OPRTDATA ALPHA USOC**!
OPRTDATA ALPHA  USLA**!
OPRTDATA ALPHA  USLN**!
OPRTDATA ALPHA  USLS**!
OPRTDATA ALPHA  USOS**!
OPRTDATA ALPHA  USOF**!
OPRTDATA ALPHA  USPD**!
```

Print data files with names containing first 4 letters shown

I-B.2.b. USGOLEXD - Enter Exogenous Data

Exogenous data is needed to drive a GOL country model. Data is needed for the desired projection horizon. Note that trade prices, price deflators, and income data are all entered in nominal terms and are converted to real terms by the structure of the model.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*************************************************************************
*
 USGOLEXD - ENTER DATA FOR EXOGENOUS VARIABLES
*
*************************************************************************
&END
DELETE DATA TIME;
DELETE DATA USICP;
DELETE DATA USINC;
DELETE DATA USPIN;
DELETE DATA USPNG;
DELETE DATA USPOP;
DELETE DATA USWIN;
DELETE DATA USPT**;
DEDIT TIME     ,1,1975;
COMMENT  TIME                                                          ;
DATA       75.000       76.000       77.000       78.000       79.000       80.000
     81.000       82.000       83.000       84.000       85.000       86.000
     87.000       88.000       89.000       90.000  ;
FILE;
DEDIT USICP    ,1,1975;
COMMENT  INDEX OF COST OF PRODUCTION (1976=100)                        ;
DATA        94.454    100.000    103.757    111.986    128.801    142.934
           155.098    166.905    179.428    192.308    205.546    218.783
           232.021    244.902    259.571    275.664    ;
FILE;
|                                    |                                    |
DEDIT USPTOO   ,1,1975;
COMMENT  PRICE (TRADE)  - OTHER OILS (US$/MT)                          ;
DATA       480.000      480.000      475.000      603.000      456.000      508.000
  426  437  468  515  559  605  654  705  762  823    ;
FILE;
DEDIT USPTDB   ,1,1975;
COMMENT  PRICE (TRADE)  - DAIRY-BUTTER (US$/MT)                        ;
DATA      1938.000     2186.000     2494.000     2373.000     2460.000     2488.000
 2641 2669 3176 3627 4058 4401 4736 5119 5550 6016    ;
FILE;
DEDIT USPTDC   ,1,1975;
COMMENT  PRICE (TRADE)  - DAIRY-CHEESE (US$/MT)                        ;
DATA      2062.000     2203.000     2281.000     2435.000     2604.000     2872.000
 3048 3081 3667 4187 4684 5080 5467 5909 6406 6945    ;
FILE;
DEDIT USPTDO   ,1,1975;
COMMENT  PRICE (TRADE)  - DAIRY-OTHER PRODUCTS (US$/MT)                ;
DATA      1243.000     1105.000     1464.000      708.000      386.000      446.000
  473  478  569  650  727  788  849  918  995 1079    ;
FILE;
OPRTDATA   TIME    ;
OPRTDATA USICP     ;
OPRTDATA USINC     ;
OPRTDATA USPIN     ;
OPRTDATA USPNG     ;
OPRTDATA USPOP     ;
OPRTDATA ALPHA  USPT**;
OPRTDATA USWIN     ;
```

I-B.2.c. USGOLEQP - Enter Export Quota Policy Data

The standard model uses export and import quotas as upper limits on exports and imports, respectively. If quotas appear in model equation, a "no quota" situation must be emulated by a quota much larger than any possible trade (or it may be set at the maximum port capacity for trade). Therefore, export quota data may be required even if export quotas, per se, are not effective.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*****************************************************************************
*
 USGOLEQP - ENTER EXPORT QUOTA POLICY DATA
*
*****************************************************************************
&END
DELETE DATA USEQ**;
DEDIT USEQBF  ,1,1975;
COMMENT  EXPORT QUOTA * BEEF+VEAL (1000 MT)                                  ;
DATA     20000.    20000.    20000.    20000.    20000.    20000.
    20000.    20000.    20000.    20000.    20000.    20000.
    20000.    20000.    20000.    20000. ;
FILE;

DEDIT USEQOO  ,1,1975;
COMMENT  EXPORT QUOTA * OTHER OILS (1000 MT)                                 ;
DATA      2000.     2000.     2000.     2000.     2000.     2000.     2000.
     2000.     2000.     2000.     2000.     2000.     2000.
     2000.     2000.     2000.     2000. ;
FILE;
DEDIT USEQDB  ,1,1975;
COMMENT  EXPORT QUOTA * DAIRY-BUTTER (1000 MT)                               ;
DATA      1000.     1000.     1000.     1000.     1000.     1000.
     1000.     1000.     1000.     1000.     1000.     1000.
     1000.     1000.     1000.     1000. ;
FILE;
DEDIT USEQDC  ,1,1975;
COMMENT  EXPORT QUOTA * DAIRY-CHEESE (1000 MT)                               ;
DATA      3000.     3000.     3000.     3000.     3000.     3000.     3000.
     3000.     3000.     3000.     3000.     3000.     3000.
     3000.     3000.     3000.     3000. ;
FILE;
DEDIT USEQDO  ,1,1975;
COMMENT  EXPORT QUOTA * DAIRY-OTHER PRODUCTS (1000 MT)                       ;
DATA      1000.     1000.     1000.     1000.     1000.     1000.     1000.
     1000.     1000.     1000.     1000.     1000.     1000.
     1000.     1000.     1000.     1000. ;
FILE;
OPRTDATA ALPHA  USEQ**;
```

I-B.2.d. USGOLMQP - Enter Import Quota Policy Data

Analogous to the case with export quotas, import quota numbers may be required for traded commodities whether or not the commodities are imported and/or have actual quotas. Import quotas, if present in equations, are prevented from affecting the standard model simply by putting in a large number; e.g., a number larger than anticipated consumption.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*****************************************************************************
*
 USGOLMQP - ENTER IMPORT QUOTA POLICY DATA
*
*****************************************************************************
&END
DELETE DATA USMQ**;
DEDIT USMQBF  ,1,1975;
COMMENT  IMPORT QUOTA * BEEF+VEAL (1000 MT)                                  ;
DATA       784.       909.       909.       981.      1025.       909.
      909.       909.       909.       909.       909.       909.
      909.       909.       909.       909. ;
FILE;
DEDIT USMQPK  ,1,1975;
COMMENT  IMPORT QUOTA * PORK (1000 MT)                                       ;
DATA     15000.     15000.     15000.     15000.     15000.     15000.
    15000.     15000.     15000.     15000.     15000.     15000.
    15000.     15000.     15000.     15000. ;
FILE;
|                                |                                          |

DEDIT USMQDB  ,1,1975;
COMMENT  IMPORT QUOTA * DAIRY-BUTTER (1000 MT)                               ;
DATA      1000.      1000.      1000.      1000.      1000.      1000.
     1000.      1000.      1000.      1000.      1000.      1000.
     1000.      1000.      1000.      1000. ;
FILE;
DEDIT USMQDC  ,1,1975;
COMMENT  IMPORT QUOTA * DAIRY-CHEESE (1000 MT)                               ;
DATA        66.        78.        78.        90.        91.        91.
       91.        84.        91.        91.        91.        91.
       91.        91.        91.        91. ;
FILE;
DEDIT USMQDO  ,1,1975;
COMMENT  IMPORT QUOTA * DAIRY-OTHER PRODUCTS (1000 MT)                        ;
DATA         3.         3.         3.         3.         3.         3.
        3.         3.         3.         3.         3.         3.
        3.         3.         3.         3. ;
FILE;
OPRTDATA ALPHA  USMQ**;
```

I-B.2.e. USGOLTEP - Enter Export Tax/Subsidy Policy Data

The standard model assumes an export tax variable for traded commodities. The variable assumes a specific (e.g. $/metric ton) tax rate. No export tax means a zero rate while an export subsidy means a negative rate. If an export tax is " ad valorem," it can be entered as an ad-valorem rate into the data file if the model equations are modified appropriately. Note that data comments can identify the data and its unit of measurement.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*************************************************************************
*
 USGOLTEP - ENTER EXPORT TAX POLICY DATA
*
*************************************************************************
&END
DELETE DATA USTE**;
DEDIT USTEBF  ,1,1975;
COMMENT  TAX(+)/SUBSIDY(-) (EXPORT) * BEEF+VEAL (US$/MT)                ;
DATA          0.          0.          0.          0.          0.          0.
        0.          0.          0.          0.          0.          0.
        0.          0.          0.          0. ;
FILE;
DEDIT USTEPK  ,1,1975;
COMMENT  TAX(+)/SUBSIDY(-) (EXPORT) * PORK (US$/MT)                     ;
DATA          0.          0.          0.          0.          0.          0.
        0.          0.          0.          0.          0.          0.
        0.          0.          0.          0. ;
FILE;
DEDIT USTEML  ,1,1975;
COMMENT  TAX(+)/SUBSIDY(-) (EXPORT) * MUTTON+LAMB (US$/MT)              ;
DATA          0.          0.          0.          0.          0.          0.
        0.          0.          0.          0.          0.          0.
        0.          0.          0.          0. ;
FILE;
|                                    |                                   |

DEDIT USTEDC  ,1,1975;
COMMENT  TAX(+)/SUBSIDY(-) (EXPORT) * DAIRY-CHEESE (US$/MT)             ;
DATA          0.          0.          0.          0.          0.          0.
        0.          0.          0.          0.          0.          0.
        0.          0.          0.          0. ;
FILE;
DEDIT USTEDO  ,1,1975;
COMMENT  TAX(+)/SUBSIDY(-) (EXPORT) * DAIRY-OTHER PRODUCTS (US$/MT)     ;
DATA      -1220.      -1280.      -1072.      -1980.      -2584.      -2584.
        0.          0.          0.          0.          0.          0.
        0.          0.          0.          0. ;
FILE;
OPRTDATA ALPHA  USTE**;
```

I-B.2.f. USGOLTMP - Enter Import Tariff/Subsidy Policy Data

The comments about export tax/subsidy data apply equally to import tax/subsidy data. Import taxes are referred to as tariffs but are generally meant to be any taxes or charges that apply to imported, as opposed to domestically produced, commodities. Subsidies are entered as negative taxes.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*****************************************************************************
*
 USGOLTMP - ENTER IMPORT TAX (TARIFF) POLICY DATA
*
*****************************************************************************
&END
DELETE DATA USTM**;
DEDIT USTMBF  ,1,1975;
COMMENT  TARIFF(+)/SUBSIDY(-) (IMPORT) * BEEF+VEAL (US$/MT)                ;
DATA        69.        69.        69.        69.        69.        57.
       46.        46.        46.        46.        46.        46.
       46.        46.        46.        46. ;
FILE;
DEDIT USTMPK  ,1,1975;
COMMENT  TARIFF(+)/SUBSIDY(-) (IMPORT) * PORK (US$/MT)                     ;
DATA        61.        61.        61.        61.        61.        61.
       61.        61.        61.        61.        61.        61.
       61.        61.        61.        61. ;
FILE;
|                                    |                                    |

DEDIT USTMDB  ,1,1975;
COMMENT  TARIFF(+)/SUBSIDY(-) (IMPORT) * DAIRY-BUTTER (US$/MT)             ;
DATA       138.       138.       138.       138.       138.       138.
      138.       138.       138.       138.       138.       138.
      138.       138.       138.       138. ;
FILE;
DEDIT USTMDC  ,1,1975;
COMMENT  TARIFF(+)/SUBSIDY(-) (IMPORT) * DAIRY-CHEESE (%)                  ;
DATA        10.        10.        10.        10.        10.        10.
       10.        10.        10.        10.        10.        10.
       10.        10.        10.        10. ;
FILE;
DEDIT USTMDO  ,1,1975;
COMMENT  TARIFF(+)/SUBSIDY(-) (IMPORT) * DAIRY-OTHER PRODUCTS (US$/MT)     ;
DATA        44.        44.        44.        44.        44.        44.
       44.        44.        44.        44.        44.        44.
       44.        44.        44.        44. ;
FILE;
OPRTDATA ALPHA  USTM**;
```

I-B.2.g. USGOLTCP - Enter Consumption Tax/Subsidy Policy Data

Consumption taxes/subsidies are those taxes applying to consumed products whether produced domestically or imported. The comments about import tax/subsidy data apply equally to this data.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*****************************************************************************
*
 USGOLTCP - ENTER CONSUMPTION TAX POLICY DATA
*
*****************************************************************************
&END
DELETE DATA USTC**;
DEDIT USTCBF  ,1,1975;
COMMENT  TAX(+)/SUBSIDY(-) (CONSUMPTION) * BEEF+VEAL (US$/MT)                ;
DATA          0.          0.          0.          0.          0.          0.
         0.          0.          0.          0.          0.          0.
         0.          0.          0.          0. ;
FILE;
DEDIT USTCPK  ,1,1975;
COMMENT  TAX(+)/SUBSIDY(-) (CONSUMPTION) * PORK (US$/MT)                     ;
DATA          0.          0.          0.          0.          0.          0.
         0.          0.          0.          0.          0.          0.
         0.          0.          0.          0. ;
FILE;
DEDIT USTCML  ,1,1975;
COMMENT  TAX(+)/SUBSIDY(-) (CONSUMPTION) * MUTTON+LAMB (US$/MT)              ;
DATA          0.          0.          0.          0.          0.          0.
         0.          0.          0.          0.          0.          0.
         0.          0.          0.          0. ;
FILE;
```

```
DEDIT USTCDC  ,1,1975;
COMMENT  TAX(+)/SUBSIDY(-) (CONSUMPTION) * DAIRY-CHEESE (US$/MT)             ;
DATA          0.          0.          0.          0.          0.          0.
         0.          0.          0.          0.          0.          0.
         0.          0.          0.          0. ;
FILE;
DEDIT USTCDO  ,1,1975;
COMMENT  TAX(+)/SUBSIDY(-) (CONSUMPTION) * DAIRY-OTHER PRODUCTS (US$/MT)     ;
DATA          0.          0.          0.          0.          0.          0.
         0.          0.          0.          0.          0.          0.
         0.          0.          0.          0. ;
FILE;
OPRTDATA ALPHA  USTC**;
```

I-B.2.h. USGOLTPP - Enter Production Tax/Subsidy Policy Data

Production tax/subsidy data is needed for commodities produced domestically. Comments applicable to other tax/subsidy data apply.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*****************************************************************************
*
 USGOLTPP - ENTER PRODUCTION TAX POLICY DATA
*
*****************************************************************************
&END
DELETE DATA USTP**;
DEDIT USTPBF  ,1,1975;
COMMENT  TAX(+)/SUBSIDY(-) (PRODUCTION) * BEEF+VEAL (US$/MT)                ;
DATA          0.          0.          0.          0.          0.          0.
         0.          0.          0.          0.          0.          0.
         0.          0.          0.          0. ;
FILE;
DEDIT USTPPK  ,1,1975;
COMMENT  TAX(+)/SUBSIDY(-) (PRODUCTION) * PORK (US$/MT)                     ;
DATA          0.          0.          0.          0.          0.          0.
         0.          0.          0.          0.          0.          0.
         0.          0.          0.          0. ;
FILE;
DEDIT USTPML  ,1,1975;
COMMENT  TAX(+)/SUBSIDY(-) (PRODUCTION) * MUTTON+LAMB (US$/MT)              ;
DATA          0.          0.          0.          0.          0.          0.
         0.          0.          0.          0.          0.          0.
         0.          0.          0.          0. ;
FILE;
|                                    |                                    |

DEDIT USTPDB  ,1,1975;
COMMENT  TAX(+)/SUBSIDY(-) (PRODUCTION) * DAIRY-BUTTER (US$/MT)             ;
DATA          0.          0.          0.          0.          0.          0.
         0.          0.          0.          0.          0.          0.
         0.          0.          0.          0. ;
FILE;
DEDIT USTPDC  ,1,1975;
COMMENT  TAX(+)/SUBSIDY(-) (PRODUCTION) * DAIRY-CHEESE (US$/MT)             ;
DATA          0.          0.          0.          0.          0.          0.
         0.          0.          0.          0.          0.          0.
         0.          0.          0.          0. ;
FILE;
DEDIT USTPDO  ,1,1975;
COMMENT  TAX(+)/SUBSIDY(-) (PRODUCTION) * DAIRY-OTHER PRODUCTS (US$/MT)     ;
DATA          0.          0.          0.          0.          0.          0.
         0.          0.          0.          0.          0.          0.
         0.          0.          0.          0. ;
FILE;
OPRTDATA ALPHA  USTP**;
```

I-B.3. Programs for Model Initialization and Updating

The groups of TROLL commands presented above help to enter a standard model, documentation, and supporting data into a TROLL file. In order to become operational, the model needs to have coefficients for the equations. The model must also be initialized, i.e. have intercepts calculated from these coefficients and data for a base year. The programs of TROLL commands that follow perform this function. They enter coefficients and parameters into the model constant file, also enter them into the elasticity matrix used to document the elasticities, coefficients, and parameters; calculate the appropriate intercepts required for model initialization; and enter the calculated intercepts into the elasticity display matrices.

Each equation group in the model which requires parameters or coefficients has a separate program. If coefficients are to be changed in the standard model, they should be changed in these programs and the programs should be re-run so that all the appropriate information about the status of the model is updated. The effective use of these types of "maintenance" programs can transform the updating of a standard model from a frustratingly complex task to a simple orderly one.

I-B.3.a. USGOLMA - Coefficients for Margin Equations

Each of the programs to enter equation coefficients/ elasticities carries out three operations. First, existing values of the coefficients in the constant files are updated by the value following the constant name (one constant per line) via TROLL CEDIT commands. Second, the intercepts of equations using these coefficients are re-calculated using TROLL DO, and DOCORE statements. Third, the matrices containing the elasticities for display purposes are updated using TROLL MATRIX manipulation capabilities. Whenever any coefficients are changed, it should be done via these programs in order that all bookkeeping and documentation operations are performed. If equations themselves are changed, the equations calculating the intercepts in these programs must be changed accordingly.

Note that if a different base year is desired for model initialization (these examples use 1979 - 1981), the year must be changed in each of these coefficient entry programs. Data must be available for all endogenous variables for the selected base year(s).

The following program enters domestic and trade margin coefficients (elasticities), updates the intercepts of the margin equations, and inserts the new elasticities and intercepts into the display matrices for the domestic and trade margins.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*****************************************************************************
USGOLMA - COEFFICIENTS FOR MARGIN EQUATIONS
*
*****************************************************************************
&END
CEDIT USGOL;
CHANGE
USMDBFPC      0.37 ,
USMDPKPC      0.27 ,
USMDMLPC      0.0  ,
USMDDMPC      0.0  ,
```

Change constants in the constant file USGOL to these values

```
USMTDOPL      0.0    ,
USMTDBPL      0.0    ,
USMTDCPL      0.0    ,
USMTDOPL      0.0    ,
 ;
FILE;
BINDVAL CONST USGOL;
DORANGE 1979 TO 1981;
DOCORE INTERCPT = USMDBF'N/(((USPNG/USPDBF'N)**USMDBFPC'C)
*((USPNG(-1)/USPDBF(-1))**USMDBFPL'C)*USPDBF'N) ;
DO USMDBFI'C = MEAN(INTERCPT);
DOCORE INTERCPT = USMDPK'N/(((USPNG/USPDPK'N)**USMDPKPC'C)
*((USPNG(-1)/USPDPK(-1))**USMDPKPL'C)*USPDPK'N) ;
DO USMDPKI'C = MEAN(INTERCPT);
DOCORE INTERCPT = USMDML'N/(((USPNG/USPDML'N)**USMDMLPC'C)
*((USPNG(-1)/USPDML(-1))**USMDMLPL'C)*USPDML'N) ;
```

Using constants in the file USGOL, calculate a constant named 'intercept' and equate it to a constant in the file

```
DOCORE INTERCPT = USMTDC'N/(((USPNG/USPDOC'N)**USMTDCPC'C)
*((USPNG(-1)/USPDOC(-1))**USMTDCPL'C)*USPDOC'N) ;
DO USMTDCI'C = MEAN(INTERCPT);
DOCORE INTERCPT = USMTDO'N/(((USPNG/USPDDO'N)**USMTDOPC'C)
*((USPNG(-1)/USPDDO(-1))**USMTDOPL'C)*USPDDO'N) ;
DO USMTDOI'C = MEAN(INTERCPT);
DORANGE;
DOCORE USMAELAS=MATREP(USMAELAS,USMDBFI'C, 1  ,1);
DOCORE USMAELAS=MATREP(USMAELAS,USMDPKI'C, 2  ,1);
DOCORE USMAELAS=MATREP(USMAELAS,USMDMLI'C, 3  ,1);
DOCORE USMAELAS=MATREP(USMAELAS,USMDDMI'C, 4  ,1);
DOCORE USMAELAS=MATREP(USMAELAS,USMDPMI'C, 5  ,1);
```

Update the constant(elasticit matrix with the new constants and equation intercepts

```
DOCORE USMAELAS=MATREP(USMAELAS,USMTDCPL'C, 18 ,6);
DOCORE USMAELAS=MATREP(USMAELAS,USMTDOPL'C, 19 ,6);
DO USMAELAS=USMAELAS;
DOCORE OPRTMAT(USMAELAS,USMAROWE'L,USMACOLE'L,0,0,-28 ,USMATITL'L,1);
```

Print the new display matrix

I-B.3.b USGOLPS - Definitions for Supply Price Equations

Definition equations in TROLL require no data for model initialization. However, some definitional data is required for the calculation of intercepts if the definitional data appears in other explanatory equations. This program of TROLL commands creates such data for the supply prices.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*************************************************************************
*
USGOLPS - DEFINITIONS FOR SUPPLY PRICE EQUATIONS
*
*************************************************************************
&END
BINDVAL CONST USGOL;
DORANGE;
DO  USPSBF=USPDBF!N=USTCBF      =USMDBF     =USTPBF      ;
DO  USPSPK=USPDPK!N=USTCPK      =USMDPK     =USTPPK      ;
DO  USPSML=USPDML!N=USTCML      =USMDML     =USTPML      ;
DO  USPSDM=USPDDM!N=USTCDM      =USMDDM     =USTPDM      ;
DO  USPSPM=USPDPM!N=USTCPM      =USMDPM     =USTPPM      ;
DO  USPSPE=USPDPE!N=USTCPE      =USMDPE     =USTPPE      ;
DO  USPSWH=USPDWH!N=USTCWH      =USMDWH     =USTPWH      ;
DO  USPSCN=USPDCN!N=USTCCN      =USMDCN     =USTPCN      ;
DO  USPSCG=USPDCG!N=USTCCG      =USMDCG     =USTPCG      ;
DO  USPSRI=USPDRI!N=USTCRI      =USMDRI     =USTPRI      ;
DO  USPSSB=USPDSB!N=USTCSB      =USMDSB     =USTPSB      ;
DO  USPSOS=USPDOS!N=USTCOS      =USMDOS     =USTPOS      ;
DO  USPSSM=USPDSM!N=USTCSM      =USMDSM     =USTPSM      ;
DO  USPSSO=USPDSO!N=USTCSO      =USMDSO     =USTPSO      ;
DO  USPSOM=USPDOM!N=USTCOM      =USMDOM     =USTPOM      ;
DO  USPSOO=USPDOO!N=USTCOO      =USMDOO     =USTPOO      ;
DO  USPSDB=USPDDB!N=USTCDB      =USMDDB     =USTPDB      ;
DO  USPSDC=USPDDC!N=USTCDC      =USMDDC     =USTPDC      ;
DO  USPSDO=USPDDO!N=USTCDO      =USMDDO     =USTPDO      ;
OPRTDATA USPS**;
```

Create (and file) new variables from existing ones

I-B.3.c. USGOLAR - Coefficients for Crop Area Equations

This program of TROLL commands enters elasticities for the crop area equations and land supply equation, updates their intercepts, and updates the corresponding display matrix.

Note that the coefficient entry lines (cards) are ordered to correspond to a stack of columns of the display matrices. Hence, the elasticities reading down the matrix columns starting with left columns corresponds to the order of the elasticity entry lines at the beginning of the program. This pattern holds for all constant entry programs.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*************************************************************************
*
USGOLAR = COEFFICIENTS FOR CROP AREA EQUATIONS
*
*************************************************************************
&END
CEDIT USGOL;
CHANGE
USARWHWH      0.15 ,
┼                                   ┼                                   ┼
USARSBOS =0.01 ,
USAROSOS  0.11 ,
 USARTTRL    0.12  ,
USARTTTR       0.00341   ,
  ;
FILE;
BINDVAL CONST USGOL;
DORANGE;
DO  USTTRL=(USPSWH     *USYDWH'N*USARWH'N
+USPSCN     *USYDCN'N*USARCN'N+USPSCG     *USYDCG'N*USARCG'N
+USPSRI     *USYDRI'N*USARRI'N+USPSSB     *USYDSB'N*USARSB'N
+USPSOS     *USYDOS'N*USAROS'N)*100/((USICP'X)*(USARWH'N+USARCN'N+USARCG'N
+USARRI'N+USARSB'N+USAROS'N));
DORANGE 1979 TO 1981 ;
DOCORE INTERCPT=USARTT/(((USTTRL     (-1))**USARTTRL'C)
*((1+USARTTTR'C)**TIME'X))  ;
DO USARTTI'C = MEAN(INTERCPT);
┼                                   ┼                                   ┼
DOCORE INTERCPT=USAROS/(1
*((USPSWH     (-1)*USYDWH'N(-1)/USICP'X(-1))**USAROSWH'C)
*((USPSCN     (-1)*USYDCN'N(-1)/USICP'X(-1))**USAROSCN'C)
*((USPSCG     (-1)*USYDCG'N(-1)/USICP'X(-1))**USAROSCG'C)
*((USPSRI     (-1)*USYDRI'N(-1)/USICP'X(-1))**USAROSRI'C)
*((USPSSB     (-1)*USYDSB'N(-1)/USICP'X(-1))**USAROSSB'C)
*((USPSOS     (-1)*USYDOS'N(-1)/USICP'X(-1))**USAROSOS'C)*USARTT'N);
DO USAROSI'C = MEAN(INTERCPT);
DORANGE ;
DOCORE USARELAS=MATREP(USARELAS,USARTTI'C, 1, 1);
DOCORE USARELAS=MATREP(USARELAS,USARWHI'C, 2, 1);
┼                                   ┼                                   ┼
DOCORE USARELAS=MATREP(USARELAS,USARCGOS'C, 4 , 7 );
DOCORE USARELAS=MATREP(USARELAS,USARRIOS'C, 5 , 7 );
DOCORE USARELAS=MATREP(USARELAS,USARSBOS'C, 6 , 7 );
DOCORE USARELAS=MATREP(USARELAS,USAROSOS'C, 7 , 7 );
DOCORE USARELAS=MATREP(USARELAS,USARTTRL'C, 1, 9);
DOCORE USARELAS=MATREP(USARELAS,USARTTTR'C, 1, 10);
DOCORE X=ROWSUM(USARELAS,0,COMBINE(2,3,4,5,6,7));
DOCORE X=-1.0*X;
DOCORE USARELAS=MATREP(USARELAS,X,0, 8);
DO USARELAS=USARELAS;
OPRTDATA USTTRL;
DOCORE OPRTMAT(USARELAS,USARROWE'L,USARCOLE'L,0,0,-26 ,USARTITL'L,1);
```

I-B.3.d. USGOLYD - Coefficients for Crop Yield Equations

This program enters elasticities for the crop yield equations, updates their intercepts, and updates the crop yield coefficient display matrix.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*****************************************************************************
*
USGOLYD - COEFFICIENTS FOR CROP YIELD EQUATIONS
*
*****************************************************************************
&END
CEDIT USGOL;
CHANGE
USYDWHWH     0.007 ,
USYDCNCN     0.114 ,
USYDSBTR       0.00977  ,
USYDOSTR       0.010999  ,
 ;
FILE;
BINDVAL CONST USGOL;
DORANGE 1979 TO 1981 ;
DOCORE INTERCPT=USYDWH/(((USPSWH     /USPIN'X)**USYDWHWH'C)
*((USARWH'N)**USYDWHAR'C)*((1+USYDWHTR'C)**TIME'X)*USWIN'X)   ;
DO USYDWHI'C = MEAN(INTERCPT);
DOCORE INTERCPT=USYDCN/(((USPSCN     /USPIN'X)**USYDCNCN'C)
*((USARCN'N)**USYDCNAR'C)*((1+USYDCNTR'C)**TIME'X)*USWIN'X)   ;
DO USYDCNI'C = MEAN(INTERCPT);
DOCORE INTERCPT=USYDCG/(((USPSCG     /USPIN'X)**USYDCGCG'C)
*((USARCG'N)**USYDCGAR'C)*((1+USYDCGTR'C)**TIME'X)*USWIN'X)   ;
DO USYDCGI'C = MEAN(INTERCPT);
DOCORE INTERCPT=USYDRI/(((USPSRI     /USPIN'X)**USYDRIRI'C)
*((USARRI'N)**USYDRIAR'C)*((1+USYDRITR'C)**TIME'X)*USWIN'X)   ;
DO USYDRII'C = MEAN(INTERCPT);
DOCORE INTERCPT=USYDSB/(((USPSSB     /USPIN'X)**USYDSBSB'C)
*((USARSB'N)**USYDSBAR'C)*((1+USYDSBTR'C)**TIME'X)*USWIN'X)   ;
DO USYDSBI'C = MEAN(INTERCPT);
DOCORE INTERCPT=USYDOS/(((USPSOS     /USPIN'X)**USYDOSOS'C)
*((USAROS'N)**USYDOSAR'C)*((1+USYDOSTR'C)**TIME'X)*USWIN'X)   ;
DO USYDOSI'C = MEAN(INTERCPT);
DORANGE ;
DOCORE USYDELAS=MATREP(USYDELAS,USYDWHI'C, 1, 1);
DOCORE USYDELAS=MATREP(USYDELAS,USYDCNI'C, 2, 1);
DOCORE USYDELAS=MATREP(USYDELAS,USYDCGI'C, 3, 1);
DOCORE USYDELAS=MATREP(USYDELAS,USYDCGTR'C, 3 ,4);
DOCORE USYDELAS=MATREP(USYDELAS,USYDRITR'C, 4 ,4);
DOCORE USYDELAS=MATREP(USYDELAS,USYDSBTR'C, 5 ,4);
DOCORE USYDELAS=MATREP(USYDELAS,USYDOSTR'C, 6 ,4);
DO USYDELAS=USYDELAS;
DOCORE OPRTMAT(USYDELAS,USYDROWE'L,USYDCOLE'L,0,0,-26 ,USYDTITL'L,1);
```

I-B.3.e. USGOLQS - Definitions for Crop Supply Quantity Equations

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*******************************************************************************
*
USGOLQS - DEFINITIONS FOR CROP SUPPLY EQUATIONS
*
*******************************************************************************
&END
DORANGE;
DO   USQSWH=USARWH   *USYDWH    ;
DO   USQSCN=USARCN   *USYDCN    ;
DO   USQSCG=USARCG   *USYDCG    ;
DO   USQSRI=USARRI   *USYDRI    ;
DO   USQSSB=USARSB   *USYDSB    ;
DO   USQSOS=USAROS   *USYDOS    ;
OPRTDATA USQSWH USQSCN USQSCG USQSRI USQSSB USQSOS;
```

TROLL

I-B.3.f. USGOLOL - Coefficients for Oilseed Product Equations

This program enters the elasticities for the oilseed product crushing equations, updates their intercepts, and updates their display matrix. As is the case with all of the coefficient entry programs, the new matrices and intercepts are displayed at the end of the program output by TROLL "OPRTMAT" commands.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*************************************************************************
*
USGOLOL - COEFFICIENTS FOR OILSEED PRODUCT EQUATIONS
*
*************************************************************************
&END
CEDIT USGOL;
CHANGE
 USQSSBSM    0.8,
 USQSOSOM    0.5,
 USQSSBSO   0.18,
 USQSOSOO   0.18,
 USQCSBPM   0.01  ,
 USQCOSPM   0.00  ,
 USQCSBTR   0.04982  ,
 USQCOSTR   0.03462  ,
 ;
FILE;
BINDVAL CONST USGOL;
DORANGE;
 DO USSBPM = (USQSSBSM'P*USPSSO + USQSSBSO'P*USPSSO);
DO USOSPM = (USQSOSOM'P*USPSOM + USQSOSOO'P*USPSOO);
DORANGE 1979 TO 1981 ;
DOCORE INTERCPT=USQCSB/(1
*((USSBPM    )**USQCSBPM'C)*((1+USQCSBTR'C)**TIME'X))   ;
DO USQCSBI'C = MEAN(INTERCPT);
DOCORE INTERCPT=USQCOS/(1
*((USOSPM    )**USQCOSPM'C)*((1+USQCOSTR'C)**TIME'X))   ;
DO USQCOSI'C = MEAN(INTERCPT);
DORANGE;
DOCORE USOLELAS=MATREP(USOLELAS,USQSSBSM'P, 1, 1);
DOCORE USOLELAS=MATREP(USOLELAS,USQSOSOM'P, 2, 1);
DOCORE USOLELAS=MATREP(USOLELAS,USQSSBSO'P, 1, 2);
DOCORE USOLELAS=MATREP(USOLELAS,USQSOSOO'P, 2, 2);
DOCORE USOLELAS=MATREP(USOLELAS,USQCSBI'C, 1, 3);
DOCORE USOLELAS=MATREP(USOLELAS,USQCOSI'C, 2, 3);
DOCORE USOLELAS=MATREP(USOLELAS,USQCSBPM'C,1,4);
DOCORE USOLELAS=MATREP(USOLELAS,USQCOSPM'C,2,4);
DOCORE USOLELAS=MATREP(USOLELAS,USQCSBTR'C,1,5);
DOCORE USOLELAS=MATREP(USOLELAS,USQCOSTR'C,2,5);
DO USOLELAS=USOLELAS;
DO  USQSSM=USQSSBSM'P*USQCSB  ;
DO  USQSSO=USQSSBSO'P*USQCSB  ;
DO  USQSOM=USQSOSOM'P*USQCOS  ;
DO  USQSOO=USQSOSOO'P*USQCOS  ;
OPRTDATA USSBPM USOSPM USQSSM USQSSO USQSOM USQSOO;
DOCORE OPRTMAT(USOLELAS,USOLROWE'L,USOLCOLE'L,0,0,-22,USOLTITL'L,1);
```

I-B.3.g. USGOLFC - Parameters for Feed Cost Equations

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*****************************************************************************
*
USGOLFC - PARAMETERS FOR FEED COST EQUATIONS
*
*****************************************************************************
&END
CEDIT USGOL;
 CHANGE
       USFCBFWH                0.01000 ,
       USFCPKWH                0.01000 ,
       USFCMLWH                0.01000 ,

       USFCPMOM                0.0     ,
       USFCPEOM                0.0     ,
 ;
FILE;
BINDVAL CONST USGOL;
DORANGE;
DOCORE USFCELAS=MATREP(USFCELAS,USFCBFWH'P, 1, 1);
DOCORE USFCELAS=MATREP(USFCELAS,USFCPKWH'P, 2, 1);
DOCORE USFCELAS=MATREP(USFCELAS,USFCMLWH'P, 3, 1);
DOCORE USFCELAS=MATREP(USFCELAS,USFCDMWH'P, 4, 1);
DOCORE USFCELAS=MATREP(USFCELAS,USFCPMWH'P, 5, 1);

DOCORE USFCELAS=MATREP(USFCELAS,USFCPMOM'P, 5, 5);
DOCORE USFCELAS=MATREP(USFCELAS,USFCPEOM'P, 6, 5);
DO USFCELAS=USFCELAS;
DO  USFCBF=USFCBFOM'P*USPDOM'N+USFCBFSM'P*USPDSM'N+
USFCBFCG'P*USPDCG'N+USFCBFCN'P*USPDCN'N+USFCBFWH'P*USPDWH'N;
DO  USFCPK=USFCPKOM'P*USPDOM'N+USFCPKSM'P*USPDSM'N+
USFCPKCG'P*USPDCG'N+USFCPKCN'P*USPDCN'N+USFCPKWH'P*USPDWH'N;
DO  USFCML=USFCMLOM'P*USPDOM'N+USFCMLSM'P*USPDSM'N+
USFCMLCG'P*USPDCG'N+USFCMLCN'P*USPDCN'N+USFCMLWH'P*USPDWH'N;
DO  USFCDM=USFCDMOM'P*USPDOM'N+USFCDMSM'P*USPDSM'N+
USFCDMCG'P*USPDCG'N+USFCDMCN'P*USPDCN'N+USFCDMWH'P*USPDWH'N;
DO  USFCPM=USFCPMOM'P*USPDOM'N+USFCPMSM'P*USPDSM'N+
USFCPMCG'P*USPDCG'N+USFCPMCN'P*USPDCN'N+USFCPMWH'P*USPDWH'N;
DO  USFCPE=USFCPEOM'P*USPDOM'N+USFCPESM'P*USPDSM'N+
USFCPECG'P*USPDCG'N+USFCPECN'P*USPDCN'N+USFCPEWH'P*USPDWH'N;
OPRTDATA USFCBF USFCPK USFCML USFCDM USFCPM USFCPE;
DOCORE OPRTMAT(USFCELAS,USFCROWE'L,USFCCOLE'L,0,0,-19,USFCTITL'L,1);
```

I-B.3.h. USGOLLP - Coefficients for Livestock Product Equations

Analagous to other constant entry programs, this program enters coefficients for all of the livestock number and supply equations.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*************************************************************************
*
USGOLLP - COEFFICIENTS FOR LIVESTOCK PRODUCT EQUATIONS
*
*************************************************************************
&END
CEDIT USGOL;
CHANGE
USLABFPC   -0.096 ,
USLSBFPC    0.142 ,
USOSBFPC    0.118 ,
|                                  |                                  |
USLNDMLG    0.964 ,
USLNPELG  0.0 ,
 ;
FILE;
BINDVAL CONST USGOL;
DORANGE 1979 TO 1981 ;
DOCORE INTERCPT=USLABF/(((USPSBF    /USFCBF    )**USLABFPC'C)
*((USPSBF    (-1)/USFCBF    (-1))**USLABFPL'C)*USLNBF'N);
DO USLABFI'C = MEAN(INTERCPT);
DOCORE INTERCPT=USLSBF/(((USPSBF    /USFCBF    )**USLSBFPC'C)
*((USPSBF    (-1)/USFCBF    (-1))**USLSBFPL'C)*USLNBF'N);
DO USLSBFI'C = MEAN(INTERCPT);
DOCORE INTERCPT=USOSBF/(((USPSBF    /USFCBF    )**USOSBFPC'C)
*((USPSBF    (-1)/USFCBF    (-1))**USOSBFPL'C)*USLSBF'N*
((1+USOSBFTR'C)**TIME'X))  ;
|                                  |                                  |
DOCORE INTERCPT=USOSPE/(((USPSPE    /USFCPE    )**USOSPEPC'C)
*((USPSPE    (-1)/USFCPE    (-1))**USOSPEPL'C)*USLNPE'N*
((1+USOSPETR'C)**TIME'X))  ;
DO USOSPEI'C = MEAN(INTERCPT);
DORANGE;
DOCORE USLPELAS=MATREP(USLPELAS,USLABFI'C,   1, 1);
DOCORE USLPELAS=MATREP(USLPELAS,USLSBFI'C,   2, 1);
DOCORE USLPELAS=MATREP(USLPELAS,USOSBFI'C,   3, 1);
|                                  |                                  |
DOCORE USLPELAS=MATREP(USLPELAS,USLNDMLG'P, 10, 5);
DOCORE USLPELAS=MATREP(USLPELAS,USLNPELG'P, 13, 5);
DO USLPELAS=USLPELAS;
DOCORE OPRTMAT(USLPELAS,USLPROWE'L,USLPCOLE'L,0,0,-28 ,USLPTITL'L,1);
```

I-B.3.i. USGOLDP - Coefficients for Dairy Product Supply Equations

This program enters elasticities for the dairy product supply equations and updates the intercepts for these equations. All appropriate display matrices are updated.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*****************************************************************************
*
USGOLDP - COEFFICIENTS FOR DAIRY PRODUCT SUPPLY EQUATIONS
*
*****************************************************************************
&END
CEDIT USGOL;
CHANGE
USQSDBDB    .50    ,
USQSDCDB   -.22    ,
USQSDODB    0.06   ,
USQSDBDC   -.70    ,
USQSDCDC    0.70   ,
USQSDODC   -1.50   ,
USQSDBDO    0.20   ,
USQSDCDO   -.48    ,
USQSDODO    1.44,
 ;
FILE;
BINDVAL CONST USGOL;
DORANGE;
DO USQMDM = USQSDM - USQDDM ;
DORANGE 1979 TO 1981 ;
DOCORE INTERCPT=USQSDB'N/(((USPSDB    /USPSDM    )**USQSDBDB'C)
*((USPSDC    /USPSDM    )**USQSDBDC'C)*((USPSDO        /USPSDM    )**USQSDBDO'C)
*USQMDM    ) ;
DO USQSDBI'C = MEAN(INTERCPT);
DOCORE INTERCPT=USQSDC'N/(((USPSDB    /USPSDM    )**USQSDCDB'C)
*((USPSDC    /USPSDM    )**USQSDCDC'C)*((USPSDO        /USPSDM    )**USQSDCDO'C)
*USQMDM    ) ;
DO USQSDCI'C = MEAN(INTERCPT);
DOCORE INTERCPT=USQSDO'N/(((USPSDB    /USPSDM    )**USQSDODB'C)
*((USPSDC    /USPSDM    )**USQSDODC'C)*((USPSDO        /USPSDM    )**USQSDODO'C)
*USQMDM    ) ;
DO USQSDOI'C = MEAN(INTERCPT);
DORANGE ;
DOCORE USDPELAS=MATREP(USDPELAS,USQSDBI'C, 1, 1);
DOCORE USDPELAS=MATREP(USDPELAS,USQSDCI'C, 2, 1);
DOCORE USDPELAS=MATREP(USDPELAS,USQSDOI'C, 3, 1);
DOCORE USDPELAS=MATREP(USDPELAS,USQSDBDB'C,1,2);
DOCORE USDPELAS=MATREP(USDPELAS,USQSDCDB'C,2,2);
DOCORE USDPELAS=MATREP(USDPELAS,USQSDODB'C,3,2);
DOCORE USDPELAS=MATREP(USDPELAS,USQSDBDC'C,1,3);
DOCORE USDPELAS=MATREP(USDPELAS,USQSDCDC'C,2,3);
DOCORE USDPELAS=MATREP(USDPELAS,USQSDODC'C,3,3);
DOCORE USDPELAS=MATREP(USDPELAS,USQSDBDO'C,1,4);
DOCORE USDPELAS=MATREP(USDPELAS,USQSDCDO'C,2,4);
DOCORE USDPELAS=MATREP(USDPELAS,USQSDODO'C,3,4);
DOCORE X = ROWSUM(USDPELAS,0,COMBINE(2,3,4));
DOCORE X=-1.0*X;
DOCORE USDPELAS = MATREP(USDPELAS,X,0,5);
DO USDPELAS=USDPELAS;
OPRTDATA USQMDM;
DOCORE OPRTMAT(USDPELAS,USDPROWE'L,USDPCOLE'L,0,0,-28 ,USDPTITL'L,1);
```

I.B.3.j. USGOLFP - Parameters for Feed Demand Equations

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*****************************************************************************
*
USGOLFP - PARAMETERS FOR FEED DEMAND EQUATIONS
*
*****************************************************************************
&END
CEDIT USGOL;
 CHANGE
       USLPWTBF                0.65000 ,
       USLPWTPK                0.08000 ,
       USLPWTML                0.01000 ,
       USLPWTDM                0.06000 ,
       USLPWTPM                0.12000 ,
       USLPWTPE                0.08000 ,
       USGCAUBF                1.02000 ,
       USGCAUPK                1.28000 ,
       USGCAUML                0.0     ,
       USGCAUDM                0.46000 ,
       USGCMUPM                0.01000 ,
       USGCAUPE                0.0     ,
 ;
FILE;
BINDVAL CONST USGOL;
DORANGE;
DO  USLPI=USLPWTBF'P*USPSBF  +USLPWTPK'P*USPSPK  +
USLPWTML'P*USPSML  +USLPWTDM'P*USPSDM  +USLPWTPM'P*USPSPM  +USLPWTPE'P*USPSPE  ;
DO  USGCAU=USGCAUBF'P*USLNBF'N+USGCAUPK'P*USLNPK'N+
USGCAUML'P*USLNML'N+USGCAUDM'P*USLNDM'N+USGCMUPM'P*USQSPM'N+USGCAUPE'P*USLNPE'N;
DOCORE USFPELAS=MATREP(USFPELAS,USLPWTBF'P, 1, 1);
DOCORE USFPELAS=MATREP(USFPELAS,USLPWTPK'P, 2, 1);
DOCORE USFPELAS=MATREP(USFPELAS,USLPWTML'P, 3, 1);
DOCORE USFPELAS=MATREP(USFPELAS,USLPWTDM'P, 4, 1);
DOCORE USFPELAS=MATREP(USFPELAS,USLPWTPM'P, 5, 1);
DOCORE USFPELAS=MATREP(USFPELAS,USLPWTPE'P, 6, 1);
DOCORE USFPELAS=MATREP(USFPELAS,USGCAUBF'P, 1, 2);
DOCORE USFPELAS=MATREP(USFPELAS,USGCAUPK'P, 2, 2);
DOCORE USFPELAS=MATREP(USFPELAS,USGCAUML'P, 3, 2);
DOCORE USFPELAS=MATREP(USFPELAS,USGCAUDM'P, 4, 2);
DOCORE USFPELAS=MATREP(USFPELAS,USGCMUPM'P, 5, 2);
DOCORE USFPELAS=MATREP(USFPELAS,USGCAUPE'P, 6, 2);
DO USFPELAS=USFPELAS;
OPRTDATA USLPI USGCAU;
DOCORE OPRTMAT(USFPELAS,USFCROWE'L,USFPCOLE'L,0,0,=26,USFPTITL'L,1);
```

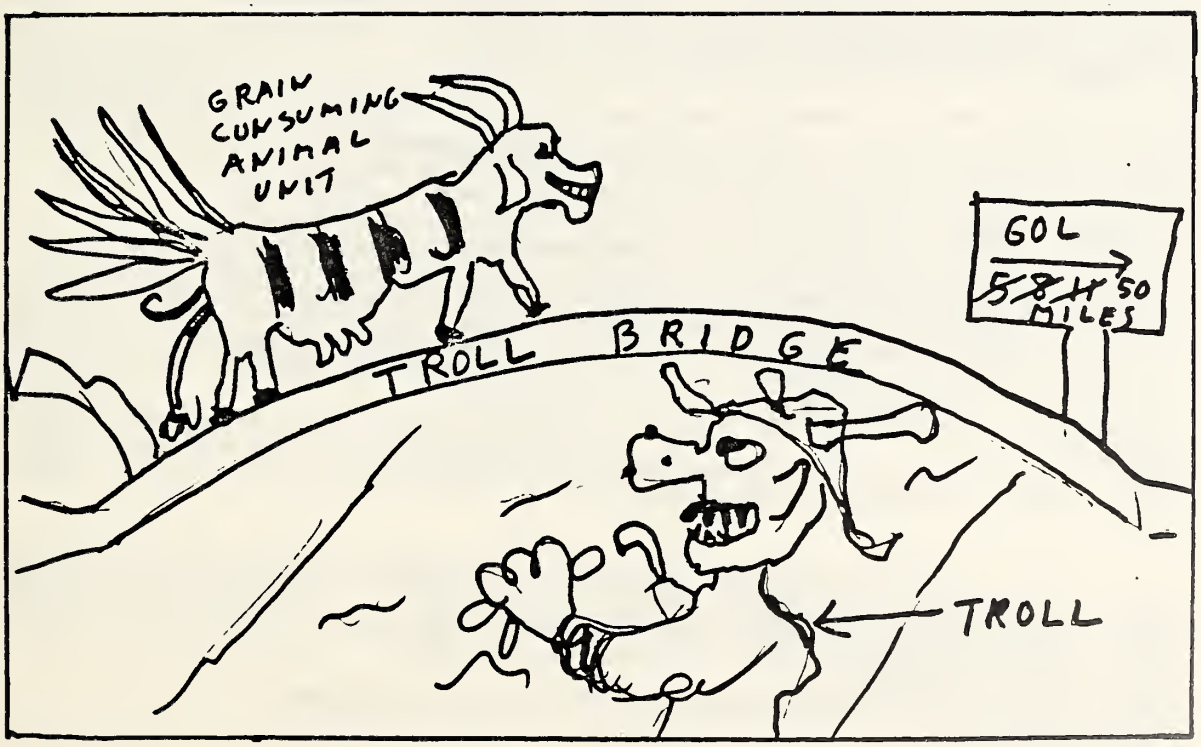
I-B.3.k. USGOLQF - Coefficients for Feed Demand Equations

This program enters feed demand elasticities and updates the appropriate intercepts and matrices.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*************************************************************************
*
USGOLQF - COEFFICIENTS FOR FEED DEMAND EQUATIONS
*
*************************************************************************
&END
CEDIT USGOL;
CHANGE
USQFWHWH     -1.00 ,
USQFCNWH      0.01 ,
USQFCGWH      0.01 ,
USQFSMWH      0.02 ,
USQFOMWH      0.02 ,
USQFWHCN      0.60 ,
|                                  |                                  |
USQFCGOM     .04     ,
USQFSMOM      0.05 ,
USQFOMOM     -1.29 ,
 ;
FILE;
BINDVAL CONST USGOL;
DORANGE 1979 TO 1981 ;
DOCORE INTERCPT=USQFWH/(((USPDWH'N/USLPI     )**USQFWHWH'C)
*((USPDCN'N/USLPI     )**USQFWHCN'C)*((USPDCG'N/USLPI     )**USQFWHCG'C)
*((USPDSM'N/USLPI     )**USQFWHSM'C)*((USPDOM'N/USLPI     )**USQFWHOM'C)
*USGCAU )  ;
|                                  |                                  |
DO USQFSMI'C = MEAN(INTERCPT);
DOCORE INTERCPT=USQFOM/(((USPDWH'N/USLPI     )**USQFOMWH'C)
*((USPDCN'N/USLPI     )**USQFOMCN'C)*((USPDCG'N/USLPI     )**USQFOMCG'C)
*((USPDSM'N/USLPI     )**USQFOMSM'C)*((USPDOM'N/USLPI     )**USQFOMOM'C)
*USGCAU )  ;
DO USQFOMI'C = MEAN(INTERCPT);
DORANGE ;
DOCORE USQFELAS=MATREP(USQFELAS,USQFWHI'C, 1, 1);
DOCORE USQFELAS=MATREP(USQFELAS,USQFCNI'C, 2, 1);
|                                  |                                  |
DOCORE USQFELAS=MATREP(USQFELAS,USQFCGOM'C, 3 , 6 );
DOCORE USQFELAS=MATREP(USQFELAS,USQFSMOM'C, 4 , 6 );
DOCORE USQFELAS=MATREP(USQFELAS,USQFOMOM'C, 5 , 6 );
DOCORE X = ROWSUM(USQFELAS, 0, COMBINE(2,3,4,5,6));
DOCORE X= -1.0*X;
DOCORE USQFELAS=MATREP(USQFELAS,X,0,7);
DO USQFELAS=USQFELAS;
DO      OPRTMAT(USQFELAS,USQFROWE'L,USQFCOLE'L,0,0,-26 ,USQFTITL'L,1);
```

I.B.3.1. USGOLQI - Coefficients for Industrial Demand Equations

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*******************************************************************************
*
USGOLQI - COEFFICIENTS FOR INDUSTRIAL DEMAND EQUATIONS
*
*******************************************************************************
&END
CEDIT USGOL;
CHANGE
USQICNCN 0.0    ,
USQICNTR   0.047  ,
 ;
FILE;
BINDVAL CONST USGOL;
DORANGE 1979 TO 1981 ;
DOCORE INTERCPT  = USQICN/   ((USPDCN'N/USPNG'X)**USQICNCN'C*
(1+USQICNTR'C)**TIME'X) ;
DO USQICNI'C = MEAN(INTERCPT);
DORANGE ;
DOCORE USQIELAS=MATREP(USQIELAS,USQICNI'C,  1,1);
DOCORE USQIELAS=MATREP(USQIELAS,USQICNCN'C,  1,2);
DOCORE USQIELAS=MATREP(USQIELAS,USQICNTR'C,  1,3);
DO USQIELAS=USQIELAS;
DO OPRTMAT(USQIELAS,USQIROWE'L,USQICOLE'L,0,0,-26,USQITITL'L,1);
```



I-B.3.m. USGOLQD - Coefficients for Food and Non-Feed Demand Equations

This program enters all of the demand elasticities and does the associated housekeeping.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*****************************************************************************
*
USGOLQD - COEFFICIENTS FOR FOOD AND NON-FEED DEMAND EQUATIONS
*
*****************************************************************************
&END
CEDIT USGOL;
CHANGE
'USQDBFBF    =.64     ,
┼                                   ┼                                   ┼
USQDDRIN     .50     ,
USQDDCIN     .30     ,
USQDDOIN     .00 ,
 ;
FILE ;
BINDVAL CONST USGOL;
DORANGE 1979 TO 1981 ;
DOCORE INTERCPT=USQDBF/(((USPDBF'N/USPNG'X)**USQDBFBF'C)
*((USPDPK'N/USPNG'X)**USQDBFPK'C)*((USPDML'N/USPNG'X)**USQDBFML'C)
*((USPDDM'N/USPNG'X)**USQDBFDM'C)*((USPDPM'N/USPNG'X)**USQDBFPM'C)
*((USPDPE'N/USPNG'X)**USQDBFPE'C)*((USPDWH'N/USPNG'X)**USQDBFWH'C)
*((USPDCN'N/USPNG'X)**USQDBFCN'C)*((USPDCG'N/USPNG'X)**USQDBFCG'C)
*((USPDRI'N/USPNG'X)**USQDBFRI'C)*((USPDSB'N/USPNG'X)**USQDBFSB'C)
*((USPDOS'N/USPNG'X)**USQDBFOS'C)*((USPDSM'N/USPNG'X)**USQDBFSM'C)
*((USPDSO'N/USPNG'X)**USQDBFSO'C)*((USPDOM'N/USPNG'X)**USQDBFOM'C)
*((USPDOO'N/USPNG'X)**USQDBFOO'C)*((USPDDB'N/USPNG'X)**USQDBFDB'C)
*((USPDDC'N/USPNG'X)**USQDBFDC'C)*((USPDDO'N/USPNG'X)**USQDBFDO'C)
*((USINC'X/(USPNG'X*USPOP'X))**USQDBFIN'C)*USPOP'X);
DO USQDBFI'C = MEAN(INTERCPT);
┼                                   ┼                                   ┼
DOCORE INTERCPT=USQDDO/(((USPDBF'N/USPNG'X)**USQDDOBF'C)
*((USPDPK'N/USPNG'X)**USQDDOPK'C)*((USPDML'N/USPNG'X)**USQDDOML'C)
*((USPDDM'N/USPNG'X)**USQDDODM'C)*((USPDPM'N/USPNG'X)**USQDDOPM'C)
*((USPDPE'N/USPNG'X)**USQDDOPE'C)*((USPDWH'N/USPNG'X)**USQDDOWH'C)
*((USPDCN'N/USPNG'X)**USQDDOCN'C)*((USPDCG'N/USPNG'X)**USQDDOCG'C)
*((USPDRI'N/USPNG'X)**USQDDORI'C)*((USPDSB'N/USPNG'X)**USQDDOSB'C)
*((USPDOS'N/USPNG'X)**USQDDOOS'C)*((USPDSM'N/USPNG'X)**USQDDOSM'C)
*((USPDSO'N/USPNG'X)**USQDDOSO'C)*((USPDOM'N/USPNG'X)**USQDDOOM'C)
*((USPDOO'N/USPNG'X)**USQDDOOO'C)*((USPDDB'N/USPNG'X)**USQDDODB'C)
*((USPDDC'N/USPNG'X)**USQDDODC'C)*((USPDDO'N/USPNG'X)**USQDDODO'C)
*((USINC'X/(USPNG'X*USPOP'X))**USQDDOIN'C)*USPOP'X);
DO USQDDOI'C = MEAN(INTERCPT);
DORANGE ;
DOCORE USQDELAS=MATREP(USQDELAS,USQDBFI'C,  1,  1);
DOCORE USQDELAS=MATREP(USQDELAS,USQDPKI'C,  2,  1);
┼                                   ┼                                   ┼
DOCORE USQDELAS=MATREP(USQDELAS, USQDDOIN'C, 16, 22);
DOCORE USQDELAS=MATREP(USQDELAS, USQDDBIN'C, 17, 22);
DOCORE USQDELAS=MATREP(USQDELAS, USQDDCIN'C, 18, 22);
DOCORE USQDELAS=MATREP(USQDELAS, USQDDOIN'C, 19, 22);
DOCORE X=ROWSUM(USQDELAS, 0,
 COMBINE(2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,22));
DOCORE X=-1.0*X;
DOCORE USQDELAS=MATREP(USQDELAS,X,0,21);
DO USQDELAS=USQDELAS;
DOCORE OPRTMAT(USQDELAS,USQDROWE'L,USQDCOLE'L,0,0,-28 ,USQDTITL'L,1);
```

I-B.3.n. USGOLSK - Coefficients for Stock Equations

This program enters stock equation elasticities and updates appropriate intercepts and the display matrix.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*****************************************************************************
*
USGOLSK = COEFFICIENTS FOR STOCK EQUATIONS
*
*****************************************************************************
&END
CEDIT USGOL;
CHANGE
USSKBFBF      0.0  ,
USSKPKPK     -0.15 ,
USSKMLML      0.0  ,
USSKPMPM      0.0  ,
┼                                  ┼                                  ┼
USSKDBDB      0.0  ,
USSKDCDC      0.0  ,
USSKDODO      0.0  ,
USSKDBPR 1.00 ,
USSKDCPR 1.00,
USSKDOPR 1.00,
   ;
FILE;
BINDVAL CONST USGOL;
DORANGE 1979 TO 1981 ;
DOCORE INTERCPT = USSKBF'N/(((USPDBF'N/USPNG'X)**USSKBFBF'C*(USQDBF'N+USQSBF'N));
DO USSKBFI'C = MEAN(INTERCPT);
DOCORE INTERCPT = USSKPK'N/(((USPDPK'N/USPNG'X)**USSKPKPK'C*(USQDPK'N+USQSPK'N));
DO USSKPKI'C = MEAN(INTERCPT);
┼                                  ┼                                  ┼
DOCORE INTERCPT = USSKDC'N/(((USPDDC'N/USPNG'X)**USSKDCDC'C*(USPRDM      /USPSDM)
 **USSKDCPR'C*(USQDDC'N+USQSDC'N));
DO USSKDCI'C = MEAN(INTERCPT);
DOCORE INTERCPT = USSKDO'N/(((USPDDO'N/USPNG'X)**USSKDODO'C*(USPRDM      /USPSDM)
 **USSKDOPR'C*(USQDDO'N+USQSDO'N));
DO USSKDOI'C = MEAN(INTERCPT);
DORANGE ;
DOCORE USSKELAS=MATREP(USSKELAS,USSKBFI'C, 1, 1);
DOCORE USSKELAS=MATREP(USSKELAS,USSKPKI'C, 2, 1);
DOCORE USSKELAS=MATREP(USSKELAS,USSKMLI'C, 3, 1);
┼                                  ┼                                  ┼
DOCORE USSKELAS=MATREP(USSKELAS,USSKDOPR'C,18, 3);
DO USSKELAS=USSKELAS;
DOCORE OPRTMAT(USSKELAS,USSKROWE'L,USSKCOLE'L,0,0,-28,USSKTITL'L,1);
```

I-B.3.o. USGOLQT - Definitions for Trade Quantity Equations

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*****************************************************************************
*
USGOLQT - DEFINITIONS FOR TRADE QUANTITY EQUATIONS
*
*****************************************************************************
&END
DORANGE ;
DO  USQTBF=USQSBF'N-USQDBF'N-(USSKBF'N-USSKBF'N(-1)) ;
DO  USQTPK=USQSPK'N-USQDPK'N-(USSKPK'N-USSKPK'N(-1)) ;
DO  USQTML=USQSML'N-USQDML'N-(USSKML'N-USSKML'N(-1)) ;
DO  USQTPM=USQSPM'N-USQDPM'N-(USSKPM'N-USSKPM'N(-1)) ;
DO  USQTPE=USQSPE'N-USQDPE'N-(USSKPE'N-USSKPE'N(-1)) ;
DO  USQTWH=USQSWH    -USQDWH'N-USQPWH'N-(USSKWH'N-USSKWH'N(-1)) ;
DO  USQTCN=USQSCN    -USQDCN'N-USQPCN'N-(USSKCN'N-USSKCN'N(-1)) ;
DO  USQTCG=USQSCG    -USQDCG'N-USQPCG'N-(USSKCG'N-USSKCG'N(-1)) ;
DO  USQTRI=USQSRI    -USQDRI'N-(USSKRI'N-USSKRI'N(-1)) ;
DO  USQTSB=USQSSB    -USQDSB'N-USQCSB'N-(USSKSB'N-USSKSB'N(-1)) ;
DO  USQTOS=USQSOS    -USQDOS'N-USQCOS'N-(USSKOS'N-USSKOS'N(-1)) ;
DO  USQTSM=USQSSM    -USQDSM'N-USQPSM'N-(USSKSM'N-USSKSM'N(-1)) ;
DO  USQTSO=USQSSO    -USQDSO'N-(USSKSO'N-USSKSO'N(-1)) ;
DO  USQTOM=USQSOM    -USQDOM'N-USQPOM'N-(USSKOM'N-USSKOM'N(-1)) ;
DO  USQTOO=USQSOO    -USQDOO'N          -(USSKOO'N-USSKOO'N(-1)) ;
DO USQTDB=USQSDB'N-USQDDB'N-(USSKDB-USSKDB(-1)) ;
DO USQTDC=USQSDC'N-USQDDC'N-(USSKDC-USSKDC(-1)) ;
DO USQTDO=USQSDO'N-USQDDO'N-(USSKDO-USSKDO(-1)) ;
OPRTDATA USQT**;
```

I-B.3.p. USGOLPE - Parameters for Demand Price Estimation Equations

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*************************************************************************
*
USGOLPE - PARAMETERS FOR DEMAND PRICE ESTIMATION EQUATIONS
*
*************************************************************************
&END
CEDIT USGOL;
CHANGE
USPEBFWQ    0.3 ,
USPEPKWQ    0.9 ,
USPEMLWQ    0.9 ,
USPEPMWQ    0.9 ,
USPEPEWQ    0.9 ,
USPEWHWQ    0.5 ,
USPECNWQ    0.9 ,
USPECGWQ    0.9 ,
USPERIWQ    0.9 ,
USPESBWQ    0.9 ,
USPEOSWQ    0.9 ,
USPESMWQ    0.9 ,
USPESOWQ    0.9 ,
USPEOMWQ    0.9 ,
USPEOOWQ    0.5 ,
USPEDBWQ    0.4 ,
USPEDCWQ    0.2 ,
USPEDOWQ    0.2 .
 ;
FILE;
BINDVAL CONST USGOL;
DORANGE ;
DOCORE USPEELAS=MATREP(USPEELAS,USPEBFWQ'P , 1, 1);
DOCORE USPEELAS=MATREP(USPEELAS,USPEPKWQ'P , 2, 1);
DOCORE USPEELAS=MATREP(USPEELAS,USPEMLWQ'P , 3, 1);
DOCORE USPEELAS=MATREP(USPEELAS,USPEPMWQ'P , 4, 1);
DOCORE USPEELAS=MATREP(USPEELAS,USPEPEWQ'P , 5, 1);
DOCORE USPEELAS=MATREP(USPEELAS,USPEWHWQ'P , 6, 1);
DOCORE USPEELAS=MATREP(USPEELAS,USPECNWQ'P , 7, 1);
DOCORE USPEELAS=MATREP(USPEELAS,USPECGWQ'P , 8, 1);
DOCORE USPEELAS=MATREP(USPEELAS,USPERIWQ'P , 9, 1);
DOCORE USPEELAS=MATREP(USPEELAS,USPESBWQ'P ,10, 1);
DOCORE USPEELAS=MATREP(USPEELAS,USPEOSWQ'P ,11, 1);
DOCORE USPEELAS=MATREP(USPEELAS,USPESMWQ'P ,12, 1);
DOCORE USPEELAS=MATREP(USPEELAS,USPESOWQ'P ,13, 1);
DOCORE USPEELAS=MATREP(USPEELAS,USPEOMWQ'P ,14, 1);
DOCORE USPEELAS=MATREP(USPEELAS,USPEOOWQ'P ,15, 1);
DOCORE USPEELAS=MATREP(USPEELAS,USPEDBWQ'P ,16, 1);
DOCORE USPEELAS=MATREP(USPEELAS,USPEDCWQ'P ,17, 1);
DOCORE USPEELAS=MATREP(USPEELAS,USPEDOWQ'P ,18, 1);
DO USPEELAS=USPEELAS;
DO OPRTMAT(USPEELAS,USQDROWE'L,USPDCOLE'L,0,0,-28,USPDTITL'L,1);
```

I-B.3.q. USGOLPD - Coefficients for Demand Price Equations

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*************************************************************************
*
USGOLPD - COEFFICIENTS FOR DEMAND PRICE EQUATIONS
*
*************************************************************************
&END
BINDVAL CONST USGOL;
DORANGE 1979 TO 1981 ;
DOCORE INTERCPT=(USPDDM'N-USTCDM      -USMDDM'N-USTPDM      )/
((USOSDB'N*USPSDB     +USOSDC'N*USPSDC     +USOSDO'N*USPSDO          )/USOMDM
DO USPDDMI'C = MEAN(INTERCPT);
DORANGE ;
DOCORE USPDELAS=MATREP(USPDELAS,USPDDMI'C,4,1);
DO USPDELAS=USPDELAS;
DO OPRTMAT(USPDELAS,USODROWE'L,USPDCOLE'L,0,0,-28,USPDTITL'L,1);
```

I-B.4. Programs for Printing Information about a Country Model

TROLL has many useful commands which can display information about models. The programs of TROLL commands presented below use various combinations of these "display" commands to produce printouts of model information. Examples of output will be illustrated.

I-B.4.a. USGOLPTM - Print a Country Model

This program produces a listing of a country model along with all of the symbol and equation comments that document the model. In addition, the coefficient display matrices are printed out. The printout of USGOL equations and comments in Roningen and Liu was produced by this program. 3/

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7;
&PRINT
*************************************************************************
*
USGOLPTM - PRINT USGOL AND ITS DISPLAY MATRICES
USMAELAS - MARGIN ELASTICITIES
USARELAS - CROP AREA ELASTICITIES
USYDELAS - CROP YIELD ELASTICITIES
USOLELAS - OILSEED PARAMETERS AND ELASTICITIES
USFCELAS - FEED COST PARAMETERS
USLPELAS - LIVESTOCK AND LIVESTOCK PRODUCT ELASTICITIES
USDPELAS - DAIRY PRODUCT ELASTICITIES
USFPELAS - FEED DEMAND PARAMETERS
USQFELAS - FEED DEMAND ELASTICITIES
USQIELAS - INDUSTRIAL DEMAND ELASTICITIES
USQDELAS - FOOD AND NON-FEED DEMAND ELASTICITIES
USSKELAS - STOCK ELASTICITIES
USPEELAS - PRICE ESTIMATE DAMPING PARAMETERS
USPDELAS - DEMAND PRICE ELASTICITIES
*
*************************************************************************
&END
USEMOD USGOL;
MODEDIT;
PRINT ALL COMMENT;
QUIT;
DO OPRTMAT(USMAELAS,USMAROWE'L,USMACOLE'L,0,0,-28 ,USMATITL'L,1);
DO OPRTMAT(USARELAS,USARROWE'L,USARCOLE'L,0,0,-26 ,USARTITL'L,1);
DO OPRTMAT(USYDELAS,USYDROWE'L,USYDCOLE'L,0,0,-26 ,USYDTITL'L,1);
DO OPRTMAT(USOLELAS,USOLROWE'L,USOLCOLE'L,0,0,-22,USOLTITL'L,1);
DOCORE OPRTMAT(USFCELAS,USFCROWE'L,USFCCOLE'L,0,0,-19,USFCTITL'L,1);
DO OPRTMAT(USLPELAS,USLPROWE'L,USLPCOLE'L,0,0,-28 ,USLPTITL'L,1);
DO OPRTMAT(USDPELAS,USDPROWE'L,USDPCOLE'L,0,0,-28 ,USDPTITL'L,1);
DOCORE OPRTMAT(USFPELAS,USFCROWE'L,USFPCOLE'L,0,0,-19,USFPTITL'L,1);
DO     OPRTMAT(USQFELAS,USQFROWE'L,USQFCOLE'L,0,0,-26 ,USQFTITL'L,1);
DO OPRTMAT(USQIELAS,USQIROWE'L,USQICOLE'L,0,0,-26,USQITITL'L,1);
DOCORE OPRTMAT(USQDELAS,USQDROWE'L,USQDCOLE'L,0,0,-28 ,USQDTITL'L,1);
DO OPRTMAT(USSKELAS,USSKROWE'L,USSKCOLE'L,0,0,-28,USSKTITL'L,1);
DO OPRTMAT(USPEELAS,USSKROWE'L,USPECOLE'L,0,0,-28,USPETITL'L,1) ;
DO OPRTMAT(USPDELAS,USQDROWE'L,USPDCOLE'L,0,0,-28,USPDTITL'L,1);
```

Print USGOL and all comments

Print matrices of constants and elasticities for USGOL

I-B.4.b. USGOLPTC - Print Matrices of Coefficients for a Country Model

This program prints all the matrices of coefficients for the model plus a listing of the current values of the coefficients in the constant file in alphabetical order by constant name.

3/ Roningen and Liu, op. cit.

```
OUTOPT RMARG 88 FPTFIELD 14 TABWIDTH 4 ;
&PRINT
*************************************************************************
*
GOLPTC    - PRINT MATRICES SUMMARIZING THE CURRENT PARAMETERIZATION OF GOL
            THESE MATRICES DISPLAY ALL OF THE ELASTICITIES, COEFFICIENTS, AND
            PARAMETERS WHICH ARE EMBEDDED IN THE MODEL EQUATIONS.  THE MATRICES:
UNITED STATES
            USMAELAS - MARGIN ELASTICITIES
            USARELAS - CROP AREA ELASTICITIES
            USYDELAS - CROP YIELD ELASTICITIES
            USOLELAS - OILSEED PARAMETERS AND ELASTICITIES
            USFCELAS - FEED COST PARAMETERS
            USLPELAS - LIVESTOCK AND LIVESTOCK PRODUCT ELASTICITIES
            USDPELAS - DAIRY PRODUCT ELASTICITIES
            USFPELAS - FEED DEMAND PARAMETERS
            USOFELAS - FEED DEMAND ELASTICITIES
            USOIELAS - INDUSTRIAL DEMAND ELASTICITIES
            USODELAS - FOOD AND NON-FEED DEMAND ELASTICITIES
            USSKELAS - STOCK ELASTICITIES
            USPEELAS - PRICE ESTIMATE DAMPING PARAMETERS
            USPOELAS - DEMAND PRICE ELASTICITIES
            USINTERC - EQUATION INTERCEPTS
REST-OF-WORLD
            RWQSELAS - SUPPLY ELASTICITIES
            RWOLELAS - OILSEED CRUSHING PARAMETERS
            RWDPELAS - DAIRY PRODUCT ELASTICITIES
            RWQDELAS - DEMAND ELASTICITIES
            RWPEELAS - PRICE (INTERNAL) ESTIMATION DAMPING PARAMETERS
WORLD MARKET CLEARING MECHANISM
            WDPEELAS - WORLD MARKET CLEARING MECHANISM PRICE DAMPING PAR
*
*************************************************************************
&END
USEMOD USGOL;
MODEDIT;
PRINT ALL COMMENT;
QUIT;
LKORD ALL;
USEMOD RWGOL;
MODEDIT;
PRINT ALL COMMENT;
QUIT;
LKORD ALL;
USEMOD WDGOL;
MODEDIT;
PRINT ALL COMMENT;
QUIT;
LKORD ALL;
LINKEDIT LKGOL;
PRINT ALL;
LKLINK;
QUIT;
DO OPRTMAT(USMAELAS,USMAROWE'L,USMACOLE'L,0,0,-28 ,USMATITL'L,1);
DO OPRTMAT(USARELAS,USARROWE'L,USARCOLE'L,0,0,-26 ,USARTITL'L,1);
DO OPRTMAT(USYDELAS,USYOROWE'L,USYDCOLE'L,0,0,-26 ,USYOTITL'L,1);
DO OPRTMAT(USOLELAS,USOLROWE'L,USOLCOLE'L,0,0,-22,USOLTITL'L,1);
DOCORE OPRTMAT(USFCELAS,USFCROWE'L,USFCCOLE'L,0,0,-19,USFCTITL'L,1);
DO OPRTMAT(USLPELAS,USLPROWE'L,USLPCOLE'L,0,0,-28 ,USLPTITL'L,1);
DO OPRTMAT(USDPELAS,USDPROWE'L,USOPCOLE'L,0,0,-28 ,USOPTITL'L,1);
DOCORE OPRTMAT(USFPELAS,USFCROWE'L,USFPCOLE'L,0,0,-19,USFPTITL'L,1);
DO     OPRTMAT(USOFELAS,USOFROWE'L,USOFCOLE'L,0,0,-26 ,USOFTITL'L,1);
DO OPRTMAT(USOIELAS,USOIROWE'L,USOICOLE'L,0,0,-26,USOITITL'L,1);
DOCORE OPRTMAT(USODELAS,USODROWE'L,USODCOLE'L,0,0,-28 ,USODTITL'L,1);
DO OPRTMAT(USSKELAS,USSKROWE'L,USSKCOLE'L,0,0,-28,USSKTITL'L,1);
DO OPRTMAT(USPEELAS,USSKROWE'L,USPECOLE'L,0,0,-28,USPETITL'L,1) ;
DO OPRTMAT(USPDELAS,USODROWE'L,USPOCOLE'L,0,0,-28,USPDTITL'L,1);
DOCORE OPRTMAT(USINTERC,USODROWE'L,USERRCOL'L,0, 0,-28,USINTERT'L,1);
DO OPRTMAT(RWQSELAS,WDROWE'L,RWQSCOLE'L,0,0,-28,RWQSTITL'L,1);
DO OPRTMAT(RWOLELAS,RWOLROWE'L,RWOLCOLE'L,0,0,-22,RWOLTITL'L,1);
DO OPRTMAT(RWDPELAS,RWOPROWE'L,RWOPCOLE'L,0,0,-28,RWOPTITL'L,1);
DO OPRTMAT(RWQDELAS,WDROWE'L,RWQDCOLE'L,0,0,-28,RWQDTITL'L,1);
DO OPRTMAT(RWPEELAS,WDROWE'L,RWPECOLE'L,0,0,-28,RWPETITL'L,1);
DO OPRTMAT(WDPEELAS,WDROWE'L,WOPECOLE'L,0,0,-28,WDPETITL'L,1);
OPRTCONST ALPHCONST          ALL,    GOL;
```

Print the model USGOL including all comments

Print the block structure of USGOL (for solution)

Quit the MODEDIT mode

Print the linkage system for GOL

Print the matrices displaying constants and parameters in GOL

Print constants in alphabetic order

I-B.4.c. USGOLPTD - Print Data for a Country Model

This program prints all of the data files associated with a country model. Note that the PRTDATA command illustrates TROLL's ability to use **'s to represent all letter combinations in a symbol.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*************************************************************************
*
USGOLPTD - PRINT ALL DATA ASSOCIATED WITH USGOL
*
*************************************************************************
&END
OPRTDATA ALPHA          US*****;
```

I-B.4.d. USGOLPTF - Print a List of Files for a Country Model

A TROLL file is a unit that holds many types of information; models, data, etc. specific to a TROLL model. TROLL manuals can be consulted for details. This program lists the names of the specific files existing in a large TROLL file.

```
LISTF ALL;
```

```
CONST_    GOL
          RWGOL
          USGOL
          WDGOL
DATA_     DATAMT2
          HRWMQWH
          HUSEQCG
          HUSMQBF1
          HUSMQBF2
          HUSMQBF3
```

I-B.4.e. USGOLPTR - Print a Cross-Reference Variable List and the Block Structure for a Country Model

For any model, TROLL can print a list which gives the number of every equation using an endogenous variable. The program below prints cross reference lists for types of variables in a standard country model. In addition, TROLL can print out the block structure and normalization used for model solution. This command will work only when the number of equations equals the number of endogenous and definitional variables in the model and therefore serves as a test of the logical completeness of a model.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*************************************************************************
*
USGOLPTR - PRINT A CROSS REFERENCE VARIABLE LIST AND A MODEL BLOCK ORDER
*
*************************************************************************
&END
USEMOD USGOL;
SYMTAB DEFINITION ENDOGENOUS EXOGENOUS;   <- Print a cross reference list for each
LKXREF ENDOGENOUS ;                          group of variables
LKXREF DEFINITION ;
LKORD ALL; <- Print the block order of the model for solution
```

I-B.5. Programs for Model Simulation

I-B.5.a. Program for Simulation of a Country Model - USGOLSIM

This program simulates a country model for 15 years. It illustrates one sequence of simulation commands. Since the program is run as a TROLL "MACRO" program, any error would normally cause execution to stop. However in this program, certain errors may occur and the program will continue executing because the macro "abort" feature has been turned off in parts of the program.

To minimize computation costs, this program may be modified. On a first run, TROLL prepares a dataset (DSET) with the same name as the model. This is a fairly expensive operation and need be done only once if the structure of the symbolic model is not to be changed. This code is saved by executing a FILEMOD statement. When the code is filed, this dataset should not be deleted. Also, simulation runs are less expensive if iteration output is not listed. However, if problems occur during simulation, TROLL has extensive diagnostic capabilities in terms of lists of iteration output. Consult the TROLL manuals for details on simulation features.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*************************************************************************
*
 USGOLSIM - SIMULATE USGOL USING A GAUSS-SEIDEL SOLUTION ROUTINE.
MATRICES GIVE INFORMATION ON THE CURRENT SIMULATION MODEL - USGOL.
USMAELAS - MARGIN ELASTICITIES
USARELAS - CROP AREA ELASTICITIES
USYDELAS - CROP YIELD ELASTICITIES
USOLELAS - OILSEED PARAMETERS AND ELASTICITIES
USFCELAS - FEED COST PARAMETERS
USLPELAS - LIVESTOCK AND LIVESTOCK PRODUCT ELASTICITIES
USDPELAS - DAIRY PRODUCT ELASTICITIES
USFPELAS - FEED DEMAND PARAMETERS
USOFELAS - FEED DEMAND ELASTICITIES
USOIELAS - INDUSTRIAL DEMAND ELASTICITIES
USODELAS - FOOD AND NON-FEED DEMAND ELASTICITIES
USSKELAS - STOCK ELASTICITIES
USPEELAS - PRICE ESTIMATE DAMPING PARAMETERS
USPDELAS - DEMAND PRICE ELASTICITIES
*
*************************************************************************
&END
PERIOD 1 ;
&ERROR &IGNORE
DELETE DSET USGOL ;
DELETE DSET USBASE ;
&ERROR &ABORT
CONOPT START 5 STOP 45 CONCR 0.1 DIVCR 300 GAMMA 10 BETA 0.7;
LKCONOPT ;
USFMOD USGOL ;
SIMULATE RELAX;
LKBINDATA ALL;
SIMSTART 1976;
SIMALG GAUSS;
FILEMOD;
&ERROR &IGNORE
ITERANGE 40 45 ;
OLIST ITERATIONS ALL;
 ;
DOSIM 15;
LKVAL INPROCESS;
&ERROR &ABORT
FILESIM USBASE ;
DO OPRTMAT(USMAELAS,USMAROWE'L,USMACOLE'L,0,0,-28 ,USMATITL'L,1);
DO OPRTMAT(USARELAS,USARROWE'L,USARCOLE'L,0,0,-26 ,USARTITL'L,1);
DO OPRTMAT(USYDELAS,USYDROWE'L,USYDCOLE'L,0,0,-26 ,USYDTITL'L,1);
DO OPRTMAT(USOLELAS,USOLROWE'L,USOLCOLE'L,0,0,-22,USOLTITL'L,1);
DOCORE OPRTMAT(USFCELAS,USFCROWE'L,USFCCOLE'L,0,0,-19,USFCTITL'L,1);
DO OPRTMAT(USLPELAS,USLPROWE'L,USLPCOLE'L,0,0,-28 ,USLPTITL'L,1);
DO OPRTMAT(USDPELAS,USDPROWE'L,USDPCOLE'L,0,0,-28 ,USDPTITL'L,1);
DOCORE OPRTMAT(USFPELAS,USFCROWE'L,USFPCOLE'L,0,0,-19,USFPTITL'L,1);
DO     OPRTMAT(USOFELAS,USOFROWE'L,USOFCOLE'L,0,0,-26 ,USOFTITL'L,1);
DO OPRTMAT(USOIELAS,USOIROWE'L,USOICOLE'L,0,0,-26,USOITITL'L,1);
DOCORE OPRTMAT(USODELAS,USODROWE'L,USODCOLE'L,0,0,-28 ,USODTITL'L,1);
DO OPRTMAT(USSKELAS,USSKROWE'L,USSKCOLE'L,0,0,-28,USSKTITL'L,1);
DO OPRTMAT(USPEELAS,USSKROWE'L,USPECOLE'L,0,0,-28,USPETITL'L,1) ;
DO OPRTMAT(USPDELAS,USODROWE'L,USPDCOLE'L,0,0,-28,USPDTITL'L,1);
```

Set the periodicity to 'annual'

Delete the old input and output DSETS

Set the convergence criteria and options

Choose the simulation technique

List all iterations after the 4oth

Simulate the model

File the output DSET

I-B.5.b. Program for Simulation of a Linked Model - LINKSIM

In TROLL it is possible to simulate a model with separate model components using TROLL's LINKMOD feature. In order to exercise this option, a "linkedit" session creates a "linkage" macro which defines the variables to be linked, the models to be used, and the solution method to be used for each model. A sample linkedit session listed below creates a linkage macro for models named USGOL, RWGOL, and WDGOL. A listing of this linkage system is given followed by a program "LINKSIM" to simulate the linked system.

```
OUTOPT PMARG 132 FPTFIELD 14 TABWIDTH 7 ;
LINKEDIT LKGOL;
VARS WDPTBF WDPTPK WDPTML WDPTPM WDPTPE
 WDPTWH WDPTCN WDPTCG WDPTRI WDPTSB WDPTOS
 WDPTSM WDPTSO WDPTOM WDPTCO
 WDPTDB WDPTDO WDPTDC
 USQTBF USQTPK USQTML USQTPM USQTPE
 USQTWH USQTCN USQTCG USQTRI USQTSB USQTOS
 USQTSM USQTSO USQTOM USQTCO
 USQTDB USQTDC USQTDO
 RWQTBF RWQTPK RWQTML RWQTPM RWQTPE
 RWQTWH RWQTCN RWQTCG RWQTRI RWQTSB RWQTOS
 RWQTSM RWQTSO RWQTOM RWQTCO
 RWQTDB RWQTDC RWQTDO
 USQSBF USQSPK USQSML USQSPM USQSPE
 USQSWH USQSCN USQSCG USQSRI USQSSB USQSOS
 USQSSM USQSSO USQSOM USQSCO
 USQSDB USQSDC USQSDO
 RWQSBF RWQSPK RWQSML RWQSPM RWQSPE
 RWQSWH RWQSCN RWQSCG RWQSRI RWQSSB RWQSOS
 RWQSSM RWQSSO RWQSOM RWQSCO
 RWQSDB RWQSDC RWQSDO
 USQDBF USQDPK USQDML USQDPM USQDPE
 USQDWH USQDCN USQDCG USQDRI USQDSB USQDOS
 USQDSM USQDSO USQDOM USQDCO
 USQDDB USQDDC USQDDO
 RWQDBF RWQDPK RWQDML RWQDPM RWQDPE
 RWQDWH RWQDCN RWQDCG RWQDRI RWQDSB RWQDOS
 RWQDSM RWQDSO RWQDOM RWQDCO
 RWQDDB RWQDDC RWQDDO
 USQFWH USQFCN USQFCG USQFSM USQFOM
 USQCSB USQCOS
 USSKBF USSKPK USSKML USSKPM USSKPE
 USSKWH USSKCN USSKCG USSKRI USSKSB USSKOS
 USSKSM USSKSO USSKOM USSKCO
 USSKDB USSKDC USSKDO ;
MODELS USGOL METHOD GAUSS, RWGOL METHOD GAUSS,
 WDGOL METHOD GAUSS;
OPRINT ALL;
OLKLINK;
FILE;
```

Declare the variables to be linked. Each must be endogenous to at least one model.

Declare the models to be linked

Print the linkage system

File the linkage system

Listing of the linkage system

```
LINKAGE SYSTEM LKGDL:

VARS:   WDPTBF  WDPTPK  WDPTML  WDPTPM  WDPTPE  WDPTWH  WDPTCN  WDPTCG  WDPTRI  WDPTSB  WDPTOS  WDPTSM  WDPTSO  WDPTOM  WDPTOO
        WDPTDB  WDPTDD  WDPTDC  USQTBF  USQTPK  USQTML  USQTPM  USQTPE  USQTWH  USQTCN  USQTCG  USQTRI  USQTSB  USQTOS  USQTSM
        USQTSO  USQTOM  USQTOO  USQTDB  USQTDC  USQTDD  RWQTBF  RWQTPK  RWQTML  RWQTPM  RWQTPE  RWQTWH  RWQTCN  RWQTCG  RWQTRI
        RWQTSB  RWQTOS  RWQTSM  RWQTSO  RWQTOM  RWQTOO  RWQTDB  RWQTDC  RWQTDD  USQSBF  USQSPK  USQSML  USQSPM  USQSPE  USQSWH
        USQSCN  USQSCG  USQSRI  USQSSB  USQSOS  USQSSM  USQSSO  USQSOM  USQSOD  USQSDB  USQSDC  USQSDO  RWQSBF  RWQSPK  RWQSML
        RWQSPM  RWQSPE  RWQSWH  RWQSCN  RWQSCG  RWQSRI  RWQSSB  RWQSOS  RWQSSM  RWQSSO  RWQSOM  RWQSOD  RWQSDB  RWQSDC  RWQSDO
        USQDBF  USQDPK  USQDML  USQDPM  USQDPE  USQDWH  USQDCN  USQDCG  USQDRI  USQDSB  USQDOS  USQDSM  USQDSD  USQDOM  USQDOD
        USQDDB  USQDDC  USQDDD  RWQDBF  RWQDPK  RWQDML  RWQDPM  RWQDPE  RWQDWH  RWQDCN  RWQDCG  RWQDRI  RWQDSB  RWQDOS  RWQDSM
        RWQDSO  RWQDOM  RWQDOO  RWQDDB  RWQDDC  RWQDDD  USQFWH  USQFCN  USQFCG  USQFSM  USQFOM  USQCSB  USQCOS  USSKBF  USSKPK
        USSKML  USSKPM  USSKPE  USSKWH  USSKCN  USSKCG  USSKRI  USSKSB  USSKOS  USSKSM  USSKSO  USSKOM  USSKOD  USSKDB  USSKDC
        USSKDD

MODELS: USGOL  METHOD GAUSS
        RWGOL  METHOD GAUSS
        WDGOL  METHOD GAUSS


LINKAGE SYSTEM LKGDL:  3 MODELS, 151 LINKAGE VARIABLES.

    USGOL
      INPUT:  WDPTBF  WDPTPK  WDPTML  WDPTPM  WDPTPE  WDPTWH  WDPTCN  WDPTCG  WDPTRI  WDPTSB  WDPTOS  WDPTSM  WDPTSO  WDPTOM  WDPTOO
                WDPTDB  WDPTDD  WDPTDC
       OUTPUT:  USQTBF  USQTPK  USQTML  USQTPM  USQTPE  USQTWH  USQTCN  USQTCG  USQTRI  USQTSB  USQTOS  USQTSM  USQTSO  USQTOM  USQTOO
                  USQTDB  USQTDC  USQTDD  USQSBF  USQSPK  USQSML  USQSPM  USQSPE  USQSWH  USQSCN  USQSCG  USQSRI  USQSSB  USQSOS
                USQSSM  USQSSO  USQSOM  USQSOD  USQSDB  USQSDC  USQSDO  USQDBF  USQDPK  USQDML  USQDPM  USQDPE  USQDWH  USQDCN  USQDCG
                  USQDRI  USQDSB  USQDOS  USQDSM  USQDSO  USQDOM  USQDOO  USQDDB  USQDDC  USQDDD  USQFWH  USQFCN  USQFCG  USQFSM
                USQFOM  USQCSB  USQCOS  USSKBF  USSKPK  USSKML  USSKPM  USSKPE  USSKWH  USSKCN  USSKCG  USSKRI  USSKSB  USSKOS  USSKSM
                  USSKSO  USSKOM  USSKOO  USSKDB  USSKDC  USSKDD

    RWGOL
      INPUT:  WDPTBF  WDPTPK  WDPTML  WDPTPM  WDPTPE  WDPTWH  WDPTCN  WDPTCG  WDPTRI  WDPTSB  WDPTOS  WDPTSM  WDPTSO  WDPTOM  WDPTOO
                WDPTDB  WDPTDD  WDPTDC
       OUTPUT:  RWQTBF  RWQTPK  RWQTML  RWQTPM  RWQTPE  RWQTWH  RWQTCN  RWQTCG  RWQTRI  RWQTSB  RWQTOS  RWQTSM  RWQTSO  RWQTOM  RWQTOO
                  RWQTDB  RWQTDC  RWQTDD  RWQSBF  RWQSPK  RWQSML  RWQSPM  RWQSPE  RWQSWH  RWQSCN  RWQSCG  RWQSRI  RWQSSB  RWQSOS
                RWQSSM  RWQSSO  RWQSOM  RWQSOD  RWQSDB  RWQSDC  RWQSDO  RWQDBF  RWQDPK  RWQDML  RWQDPM  RWQDPE  RWQDWH  RWQDCN  RWQDCG
                  RWQDRI  RWQDSB  RWQDOS  RWQDSM  RWQDSO  RWQDOM  RWQDOO  RWQDDB  RWQDDC  RWQDDD

    WDGCL
      INPUT:  USQTBF  USQTPK  USQTML  USQTPM  USQTPE  USQTWH  USQTCN  USQTCG  USQTRI  USQTSB  USQTDS  USQTSM  USQTSO  USQTOM  USQTOO
                USQTDB  USQTDC  USQTDD  RWQTBF  RWQTPK  RWQTML  RWQTPM  RWQTPE  RWQTWH  RWQTCN  RWQTCG  RWQTRI  RWQTSB  RWQTDS
              RWQTSM  RWQTSO  RWQTOM  RWQTOO  RWQTDB  RWQTDC  RWQTDD  USQSBF  USQSPK  USQSML  USQSPM  USQSPE  USQSWH  USQSCN  USQSCG
                USQSRI  USQSSB  USQSDS  USQSSM  USQSSO  USQSOM  USQSDO  USQSDB  USQSDC  USQSDD  RWQSBF  RWQSPK  RWQSML  RWQSPM
              RWQSPE  RWQSWH  RWQSCN  RWQSCG  RWQSRI  RWQSSB  RWQSDS  RWQSSM  RWQSSO  RWQSOM  RWQSDD  RWQSDB  RWQSDC  RWQSDO  USQDBF
                USQDPK  USQDML  USQDPM  USQDPE  USQDWH  USQDCN  USQDCG  USQDRI  USQDSB  USQDDS  USQDSM  USQDSD  USQDDM  USQDDD
              USQDDB  USQDDC  USQDDO  RWQDBF  RWQDPK  RWQDML  RWQDPM  RWQDPE  RWQDWH  RWQDCN  RWQDCG  RWQDRI  RWQDSB  RWQDOS  RWQDSM
                RWQDSO  RWQDOM  RWQDOO  RWQDDB  RWQDDC  RWQDDD  USQFWH  USQFCN  USQFCG  USQFSM  USQFOM  USQCSB  USQCOS  USSKBF
              USSKPK  USSKML  USSKPM  USSKPE  USSKWH  USSKCN  USSKCG  USSKRI  USSKSB  USSKOS  USSKSM  USSKSO  USSKOM  USSKOD  USSKDB
                USSKDC  USSKDD
       OUTPUT:  WDPTBF  WDPTPK  WDPTML  WDPTPM  WDPTPE  WDPTWH  WDPTCN  WDPTCG  WDPTRI  WDPTSB  WDPTOS  WDPTSM  WDPTSD  WDPTOM  WDPTOO
                  WDPTDB  WDPTDD  WDPTDC
```

[redacted].[redacted].LINKSIM.DATA

```
&ERROR &IGNORE
DELETE DSET LKUSBASE;
DELETE DSET LKRWBASE;
DELETE DSET LKWDBASE;
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
BINDVAL CONST GOL;
PERIOD 1;
CONOPT START 9 STOP 30 CONCR 0.4 DIVCR 100 GAMMA 10 ;
LKCONOPT;
LINKSIM LKGOL;
LINKOPT START 9 STOP 30 CONCR 0.35 DIVCR 100 STOPCONV ;
LKOPT ;
LKDSETS ;
SIMSTART 1976 ;
OLIST WDGOL ITERATION, ENDOGENOUS;
OLIST RWGOL ITERATION, ENDOGENOUS;
OLIST USGOL ITERATION, ENDOGENOUS;
ITERANGE WDGOL 27;
ITERANGE RWGOL 27;
ITERANGE USGOL 27;
DOTIL 1990;
LKVAL USGOL INPROCESS;
LKVAL RWGOL INPROCESS;
LKVAL WDGOL INPROCESS;
LKVAL WDGOL ITERATION, WDPTDO WDPTDO(-1);
RENAME DSET OUTPUT_USGOL LKUSBASE;
RENAME DSET OUTPUT_RWGOL LKRWBASE;
RENAME DSET OUTPUT_WDGOL LKWDBASE;
CHMODE PERM DSET ****BASE;
OPRTDSET VALUE ER PCER , VARIABLES
 WDPTBF WDPTPK WDPTML WDPTPM WDPTPE
 WDPTWH WDPTCN WDPTCG WDPTRI WDPTSB WDPTOS
 WDPTSM WDPTSO WDPTOM WDPTOO
 WDPTDB WDPTDC WDPTDO ,
RANGE 1976 TO 1990,
DSETS GOL WDBASE LKWDBASE GOLBASE;
OPRTDSET VALUE ER PCER, VARIABLES
 USQTBF USQTPK USQTML USQTPM USQTPE
 USQTWH USQTCN USQTCG USQTRI USQTSB USQTOS
 USQTSM USQTSO USQTOM USQTOO
 USQTDB USQTDC USQTDO ,
RANGE 1976 TO 1990,
DSETS GOL USBASE LKUSBASE GOLBASE;
OPRTDSET VALUE ER PCER, VARIABLES
RWQTWH RWQTCN RWQTCG RWQTRI RWQTSB RWQTOS
RWQTSM RWQTSO RWQTOM RWQTOO
RWQTDB RWQTDC RWQTDO
RWPIBF RWPIPK RWPIML RWPIPM RWPIPE
RWPIWH RWPICN RWPICG RWPIRI RWPISB RWPIOS
RWPISM RWPISO RWPIOM RWPIOO
RWPIDB RWPIDC RWPIDO,
RANGE 1976 TO 1990,
DSETS GOL RWBASE LKRWBASE GOLBASE;
LISTF DSET>;
```

Delete old model DSETs

Select convergence options for model variables

Select convergence option for linked variables

List solution iterations

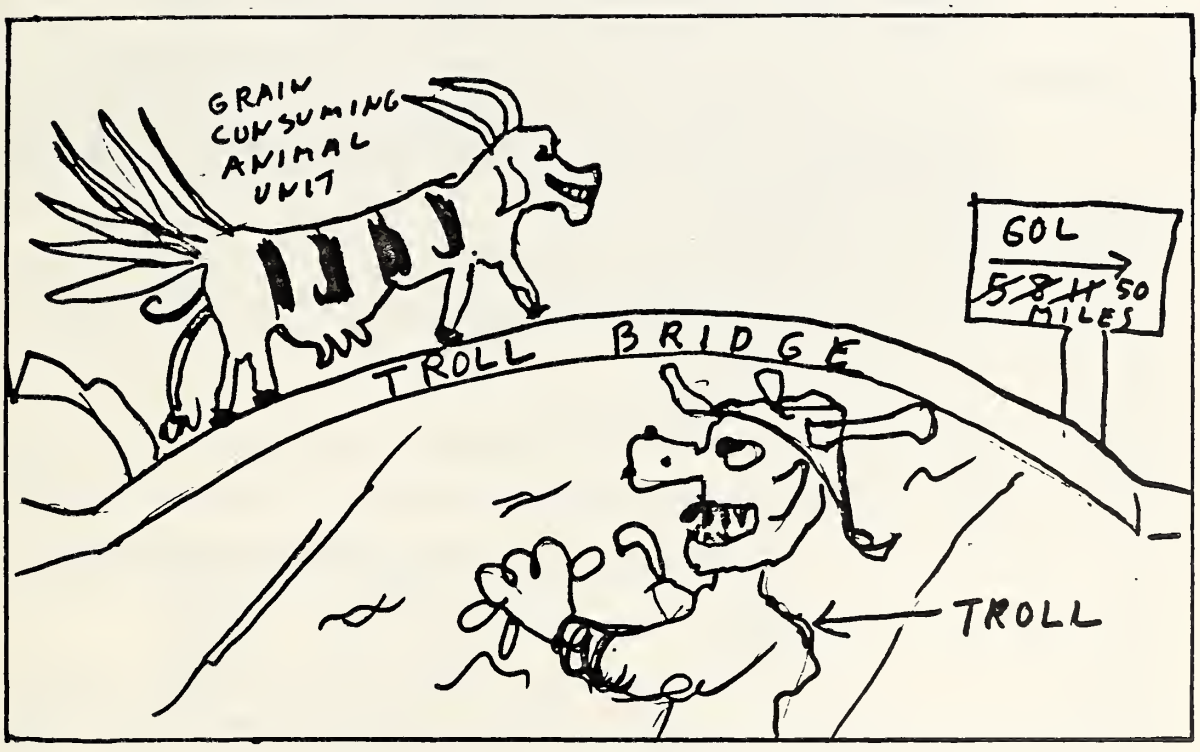
Simulate the linked model

Rename and permanently file the output DSETs

Print values of selected output variables

I-B.6. Programs for Model Validation

Model validation is an important step in the preparation of a model for actual use. The standard country model can be validated provided historical data is available in data files. Two TROLL programs aid in the validation effort.



I-B.6.a. USGOLPTE - Print Summary Matrices of Model Validation Statistics

This program prints matrices of statistics allowing a quick comparison of a simulation run to a historical data set. Comparisons are made only for the endogenous variables. The matrices present the mean and absolute mean of the error and percentage errors which result from comparing a base run to historical data.

The program is followed by a partial listing of the program which can create these display matrices.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*******************************************************************************
*
 USGOLPTE - PRINT THE SUMMARY MATRICES FOR A VALIDATION RUN OF USGOL.
            THE MATRICES ARE:
            USERRMA = MEAN ABSOLUTE ERROR OF VALIDATION RUN.
            USERRMN = MEAN OF ERROR OF VALIDATION RUN.
            USPERMA = MEAN ABSOLUTE PERCENT ERROR OF VALIDATION RUN.
            USPERMN = MEAN OF PERCENT ERROR OF VALIDATION RUN.
*
*******************************************************************************
*
          - PRINT MATRICES SUMMARIZING THE CURRENT PARAMETERIZATION OF GOL.
            THESE MATRICES DISPLAY ALL OF THE ELASTICITIES, COEFFICIENTS, AND
            PARAMETERS WHICH ARE EMBEDDED IN THE MODEL EQUATIONS.  THE MATRICES:
USMAELAS - MARGIN ELASTICITIES
USARELAS - CROP AREA ELASTICITIES
USYDELAS - CROP YIELD ELASTICITIES
USOLELAS - OILSEED PARAMETERS AND ELASTICITIES
USFCELAS - FEED COST PARAMETERS
USLPELAS - LIVESTOCK AND LIVESTOCK PRODUCT ELASTICITIES
USDPELAS - DAIRY PRODUCT ELASTICITIES
USFPELAS - FEED DEMAND PARAMETERS
USQFELAS - FEED DEMAND ELASTICITIES
USQIELAS - INDUSTRIAL DEMAND ELASTICITIES
USQDELAS - FOOD AND NON-FEED DEMAND ELASTICITIES
USSKELAS - STOCK ELASTICITIES
USPEELAS - PRICE ESTIMATE DAMPING PARAMETERS
USPDELAS - DEMAND PRICE ELASTICITIES
*
*******************************************************************************
&END
DELETE DATA US****_USBASE_ER;
DELETE DATA US****_USBASE_PCER;
PERIOD 1;
CRDATA ER PCER, VARIABLES ENDOGENOUS, RANGE 1976 TO 1980, DSETS USGOL USBASE ;
DORANGE;
DOCORE USERRMA = CRMAT( 19, 16, NA);
DOCORE USERRMN = CRMAT( 19, 16, NA);
DOCORE USPERMA=CRMAT(19,16,NA);
DOCORE USPERMN=CRMAT(19,16,NA);
DOCORE USERRMA=MATREP(USERRMA,MEAN(ABSV(USARWH_USBASE_ER)), 7, 1);
DOCORE USERRMA=MATREP(USERRMA,MEAN(ABSV(USARCN_USBASE_ER)), 8, 1);
DOCORE USERRMA=MATREP(USERRMA,MEAN(ABSV(USARCG_USBASE_ER)), 9, 1);
```

```
DOCORE USPERMN=MATREP(USPERMN,MEAN(USGTDO_USBASE_PCER),19,14);
DOCORE USPERMN=MATREP(USPERMN,MEAN(USGICN_USBASE_PCER),  8  ,15);
DOCORE USPERMN=MATREP(USPERMN,MEAN(USPTBF_USBASE_PCER),  1  ,16);
DOCORE USPERMN=MATREP(USPERMN,MEAN(USPTPK_USBASE_PCER),  2  ,16);
DOCORE USPERMN=MATREP(USPERMN,MEAN(USPTML_USBASE_PCER),  3  ,16);
DOCORE USPERMN=MATREP(USPERMN,MEAN(USPTDM_USBASE_PCER),  4  ,16);
DOCORE USPERMN=MATREP(USPERMN,MEAN(USPTPM_USBASE_PCER),  5  ,16);
DOCORE USPERMN=MATREP(USPERMN,MEAN(USPTPE_USBASE_PCER),  6  ,16);
DOCORE USPERMN=MATREP(USPERMN,MEAN(USPTWH_USBASE_PCER),  7  ,16);
DOCORE USPERMN=MATREP(USPERMN,MEAN(USPTCN_USBASE_PCER),  8  ,16);
DOCORE USPERMN=MATREP(USPERMN,MEAN(USPTCG_USBASE_PCER),  9  ,16);
DOCORE USPERMN=MATREP(USPERMN,MEAN(USPTRI_USBASE_PCER),  10 ,16);
DOCORE USPERMN=MATREP(USPERMN,MEAN(USPTSB_USBASE_PCER),  11 ,16);
DOCORE USPERMN=MATREP(USPERMN,MEAN(USPTOS_USBASE_PCER),  12 ,16);
DOCORE USPERMN=MATREP(USPERMN,MEAN(USPTSM_USBASE_PCER),  13 ,16);
DOCORE USPERMN=MATREP(USPERMN,MEAN(USPTSO_USBASE_PCER),  14 ,16);
DOCORE USPERMN=MATREP(USPERMN,MEAN(USPTOM_USBASE_PCER),  15 ,16);
DOCORE USPERMN=MATREP(USPERMN,MEAN(USPTOO_USBASE_PCER),  16 ,16);
DOCORE USPERMN=MATREP(USPERMN,MEAN(USPTOB_USBASE_PCER),  17 ,16);
DOCORE USPERMN=MATREP(USPERMN,MEAN(USPTOC_USBASE_PCER),  18 ,16);
DOCORE USPERMN=MATREP(USPERMN,MEAN(USPTDO_USBASE_PCER),  19 ,16);
DO USPERMN=USPERMN;
DOCORE      PRINT(MEAN(ABSV(USARTT_USBASE_ER)));
DOCORE      PRINT(MEAN(USARTT_USBASE_ER));
DOCORE      PRINT(MEAN(ABSV(USARTT_USBASE_PCER)));
DOCORE      PRINT(MEAN(USARTT_USBASE_PCER));
DO OPRTMAT(USERRMA,USQDROWE'L,USERRCOL'L,0,0,-28,USEATITL'L,1);
DO OPRTMAT(USERRMN,USQDROWE'L,USERRCOL'L,0,0,-28,USENTITL'L,1);
DO OPRTMAT(USPERMA,USQDROWE'L,USERRCOL'L,0,0,-28,USPATITL'L,1);
DO OPRTMAT(USPERMN,USQDROWE'L,USERRCOL'L,0,0,-28,USPNTITL'L,1);
DO OPRTMAT(USMAELAS,USMAROWE'L,USMACOLE'L,0,0,-28 ,USMATITL'L,1);
DO OPRTMAT(USARELAS,USARROWE'L,USARCOLE'L,0,0,-26 ,USARTITL'L,1);
DO OPRTMAT(USYDELAS,USYDROWE'L,USYDCOLE'L,0,0,-26 ,USYDTITL'L,1);
DO OPRTMAT(USOLELAS,USOLROWE'L,USOLCOLE'L,0,0,-22,USOLTITL'L,1);
DOCORE OPRTMAT(USFCELAS,USFCROWE'L,USFCCOLE'L,0,0,-19,USFCTITL'L,1);
DO OPRTMAT(USLPELAS,USLPROWE'L,USLPCOLE'L,0,0,-28 ,USLPTITL'L,1);
DO OPRTMAT(USDPELAS,USDPROWE'L,USDPCOLE'L,0,0,-28 ,USDPTITL'L,1);
DOCORE OPRTMAT(USFPELAS,USFCROWE'L,USFPCOLE'L,0,0,-19,USFPTITL'L,1);
DO     OPRTMAT(USQFELAS,USQFROWE'L,USQFCOLE'L,0,0,-26 ,USQFTITL'L,1);
DO OPRTMAT(USQIELAS,USQIROWE'L,USQICOLE'L,0,0,-26,USQITITL'L,1);
DOCORE OPRTMAT(USQDELAS,USQDROWE'L,USQDCOLE'L,0,0,-28 ,USCDTITL'L,1);
DO OPRTMAT(USSKELAS,USSKROWE'L,USSKCOLE'L,0,0,-28,USSKTITL'L,1);
DO OPRTMAT(USPEELAS,USSKROWE'L,USPECOLE'L,0,0,-28,USPETITL'L,1) ;
DO OPRTMAT(USPDELAS,USQDROWE'L,USPDCOLE'L,0,0,-28,USPDTITL'L,1);



OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&PRINT
*****************************************************************************
*
USGOLCSM    CREATE SUMMARY MATRICES FOR MEAN ABSOLUTE AND ADDITIVE ERRS,
OF A MODEL VALIDATION OF USGOL.  THE MATRICES ARE:
            USERRMA = MEAN ABSOLUTE  ERROR OF VALIDATION RUN,
            USERRMN = MEAN OF ERROR OF VALIDATION RUN,
            USPERMA = MEAN ABSOLUTE PERCENT ERROR OF VALIDATION RUN,
            USPERMN = MEAN OF PERCENT ERROR OF VALIDATION RUN,
*
CREATE MATRICES TO SUMMARIZE THE DIFFERENCES BETWEEN A BASELINE
RUN OF USGOL (USBASE) AND AN ALTERNATE RUN (USPROJ).  THE MATRICES:
            USPROJER = MEAN DIFFERENCE BETWEEN VALUES OF USPROJ AND USBASE,
            USPROJPR = MEAN PERCENTAGE DIFFERENCE BETWEEN VALUES OF USPROJ AND
                       USBASE,
*
*****************************************************************************
&END
DELETE DATA USERRMA;
DO     USERRMA = CRMAT( 19, 16, NA);
DELETE DATA USERRMN;
DO     USERRMN = CRMAT( 19, 16, NA);
DELETE DATA USPERMA;
DO     USPERMA=CRMAT(19,16,NA);
DELETE DATA USPERMN;
DO     USPERMN=CRMAT(19,16,NA);
```

```
DELETE GENERAL LABEL_USEATITL;
LEDIT USEATITL;
ADD TOP,     MEAN ABSOLUTE ERROR - UNITED STATES;
FILE ;
DELETE GENERAL LABEL_USENTITL;
LEDIT USENTITL;
ADD TOP, MEAN OF ERROR - UNITED STATES;
FILE ;
DELETE GENERAL LABEL_USPATITL;
LEDIT USPATITL;
ADD TOP, AVERAGE ABSOLUTE PERCENTAGE ERROR - UNITED STATES;
FILE ;
DELETE GENERAL LABEL_USPNTITL;
LEDIT USPNTITL;
ADD TOP, MEAN OF PERCENTAGE ERROR - UNITED STATES;
FILE ;
DO OPRTMAT(USERRMA,USQDROWE'L,USERRCOL'L,0,0,-28,USPATITL'L,1);
DO OPRTMAT(USERRMN,USQDROWE'L,USERRCOL'L,0,0,-28,USENTITL'L,1);
DO OPRTMAT(USPERMA,USQDROWE'L,USERRCOL'L,0,0,-28,USPATITL'L,1);
DO OPRTMAT(USPERMN,USQDROWE'L,USERRCOL'L,0,0,-28,USPNTITL'L,1);
DELETE DATA USPROJER ;
DO      USPROJER=CRMAT(19,21,NA) ;
DELETE DATA USPROJPR;
DO      USPROJPR=CRMAT(19,21,NA) ;
DELETE GENERAL LABEL_USPROCOL;
LEDIT USPROCOL;
ADD TOP,
AREA (1000 HECTARES),
YIELD (METRIC TONS / HECTARE),
LIVESTOCK NUMBERS (1000),
LIVESTOCK ADDITIONS (1000),
LIVESTOCK SLAUGHTER (1000),
QUANTITY SUPPLIED (1000 METRIC TONS),
SUPPLY PRICE (US$ / METRIC TON),
FEED COST (US$ / METRIC TON),
QUANTITY FED (1000 METRIC TONS),
QUANTITY CRUSHED (1000 METRIC TONS),
QUANTITY DEMANDED (1000 METRIC TONS),
DEMAND PRICE (US$ / METRIC TON),
ENDING STOCKS (1000 METRIC TONS),
DOMESTIC MARGIN (US$ / METRIC TON),
TRADE MARGIN (US$ / METRIC TON),
QUANTITY TRADED (1000 METRIC TONS),
PRODUCER REVENUE (1000 US$),
CONSUMPTION EXPENDITURE (1000 US$),
TRADE VALUE (1000 US$),
RATIO OF TRADE TO DOMESTIC CONSUMPTION (%),
RATIO OF TRADE TO DOMESTIC PRODUCTION (%);
FILE;
DELETE GENERAL LABEL_USPROJET;
LEDIT USPROJET;
ADD TOP, MEAN DIFFERENCE BETWEEN PROJ. AND BASE VALUES - UNITED STATES,
MEAN OF (PROJECTION VALUES - BASELINE VALUES) ;
FILE;
DELETE GENERAL LABEL_USPROJPT;
LEDIT USPROJPT;
ADD TOP, MEAN % DIFF. BETWEEN PROJ. AND BASE VALUES - UNITED STATES,
MEAN OF (% DIFFERENCE BETWEEN PROJECTION AND BASELINE VALUES) ;
FILE;
DO OPRTMAT(USPROJER,USQDROWE'L,USPROCOL'L,0,0,-28,USPROJET'L,1);
DO OPRTMAT(USPROJPR,USQDROWE'L,USPROCOL'L,0,0,-28,USPROJPT'L,1);
DELETE GENERAL LABEL_USERRCOL;
LEDIT USERRCOL ;
ADD TOP, AREA (AR), YIELD (YD), LIVESTOCK NUMBERS (LN),
 LIVESTOCK ADDITIONS (LA), LIVESTOCK SLAUGHTER (LS), QUANTITY SUPPLIED (QS),
PRICE (DEMAND) (PD), QUANTITY FED (QF), QUANTITY CRUSHED (QC),
QUANTITY DEMANDED (QD), STOCKS (SK), MARGIN (DOMESTIC) (MD),
MARGIN (TRADE) (MT), QUANTITY TRADED (QT),
QUANTITY INDUSTRIAL DEMAND (QI), PRICE (TRADE (PT) ;
FILE ;
```

I-B.6.b. Print Validation Statistics Using the OPRTDSET Commands

The OPRTDSET command can compare data in data sets. By comparing a simulation output data set with a historical data set, validation statistics can be produced. The STATS option must be used to produce the validation statistics.

```
OPRTDSET VALUE ER PCER STATS,
VARIABLES RWQTBF RWQTPK RWQTML RWQTPM RWQTPE
RWQTWH RWQTCN RWQTCG RWQTRI RWQTSB RWQTOS
RWQTSM RWQTSO RWQTOM RWQTOO
RWQTDB RWQTDC RWQTDO
RWPIBF RWPIPK RWPIML RWPIPM RWPIPE
RWPIWH RWPICN RWPICG RWPIRI RWPISB RWPIOS
RWPISM RWPISO RWPIOM RWPIOO
RWPIDB RWPIDC RWPIDO,
RANGE 1976 TO 1981, DSETS RWGOL RWBASE;
```

SIMULATION OUTPUT BY VARIABLE

RWQTCN = ENDOGENOUS

	RWTEST	RWBASE	RWBASE_ER	RWBASE_PCER
1976	-42707.1	-42707.2	-0.0625	0.000146
1977	-49339.2	-55551.6	-6212.44	12.5913
1978	-54090.1	-62702.5	-8606.44	15.9095
1979	-61807.9	-72323.3	-10515.4	17.013
1980	-64797.9	-84335.1	-19537.2	30.151

SUMMARY STATISTICS

	RWTEST	RWBASE	RWBASE_ER	RWBASE_PCER
MEAN	-54549.6	-65523.9	-8974.3	15.133
RMS	55142.8	65091.5	10999.4	17.9453
STD.DEV	9016.16	15875.3	7110.72	10.7835

I-B.7. Programs for Summarizing the Impact of Alternative Projections

An acceptable standard model will be adjusted and modified to produce a "best effort" baseline projection. Then alternative scenarios with altered policies will be run. A major aim of this type of model is to quickly examine the differences between those alternative projections and an acceptable baseline. The TROLL programs presented below allow a quick comparsion of an alternative projection with a base projection.

I-B.7.a. OPRTDSET - Print Information Comparing Output Data Sets

The TROLL OPRTDSET command can be used to print differences (and summary statistics of those differences) between alternative output data sets. An example of the use of this command is shown below.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
OPRTDSET VALUE ER PCER ; VARIABLES
 WDPTBF WDPTPK WDPTML WDPTPM WDPTPE
 WDPTWH WDPTCN WDPTCG WDPTRI WDPTSB WDPTOS
 WDPTSM WDPISU WDPTOM WDPTOO
 WDPTDB WDPTDC WDPTDO ;
RANGE 1976 TO 1990; DSETS WDGOL WDBASE GOLBASE;
OPRTDSET VALUE ER PCER ; VARIABLES
 USQTBF USQTPK USQTML USQTPM USQTPE
 USQTWH USQTCN USQTCG USQTRI USQTSB USQTOS
 USQTSM USQTSU USQTOM USQTOO
 USQTDB USQTDC USQTDO ;
RANGE 1976 TO 1990 ; DSETS USGOL USBASE GOLBASE ;
OPRTDSET VALUE ER PCER; VARIABLES
RWQTWH RWQTCN RWQTCG RWQTRI RWQTSB RWQTOS
RWQTSM RWQTSO RWQTOM RWQTOO
RWQTDB RWQTDC RWQTDO
RWPIBF RWPIPK RWPIML RWPIPM RWPIPE
RWPIWH RWPICN RWPICG RWPIRI RWPISB RWPIOS
RWPISM RWPISO RWPIOM RWPIOO
RWPIDB RWPIDC RWPIDO;
RANGE 1976 TO 1990; DSETS RWGOL RWBASE GOLBASE ;
LISTF DSET>;
```

I-B.7.b. USGOLPPJ - Print Matrices Summarizing the Difference Between an Alternative Projection and a Baseline

This program calculates a limited set of "difference" statistics and arranges them in an easily readable matrix form which is derived from the model structure. It complements a printout of more detailed information on differences between alternative projections and a baseline which can be obtained using TROLL OPRTDSET commands.

Note that OPRTDSET commands are relatively cheap to use but do not put the results in as convenient a form as USGOLPPJ. Part of the expense of this program involves the use of an expensive TROLL CRDATA command to create variables from a DSET.

A set of TROLL commands which creates the "display" matices is listed following by USGOLPPJ.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7;
&PRINT
*************************************************************************
*
 USGOLPPJ - PRINT  MATRICES TO SUMMARIZE THE DIFFERENCES BETWEEN A BASELINE
            RUN OF USGOL (USBASE) AND AN ALTERNATE RUN (USPROJ).  THE MATRICES:
            USPROJER = MEAN DIFFERENCE BETWEEN VALUES OF USPROJ AND USBASE.
            USPROJPR = MEAN PERCENTAGE DIFFERENCE BETWEEN VALUES OF USPROJ AND
                       USBASE.
*
*************************************************************************
&END
DELETE DATA PRREV**;
DELETE DATA CNEXP**;
DELETE DATA TDVAL**;
DELETE DATA RTDCN**;
DELETE DATA RTDPN**;
DELETE DATA US****_USPROJ;
DELETE DATA US****_USPROJ_ER;
DELETE DATA US****_USPROJ_PCER;
PERIOD 1;
CRDATA VALUE ER PCER, VARIABLES ENDOGENOUS DEFINITION,
RANGE 1986 TO 1990, DSETS USBASE USPROJ ;
CRDATA VALUE ER PCER, VARIABLES USPTBF USPTPK USPTML USPTFM USPTPE USPTWH USPTCN
 USPTCG USPTRI USPTSB USPTOS USPTSM USPTSO USPTUM USPTDO USPTDB USPTDC USPTDO,
RANGE 1986 TO 1990, DSETS USBASE USPROJ;
DOCORE USPROJER=MATREP(USPROJER,MEAN(USARWH_USPROJ_ER), 7, 1);
DOCORE USPROJER=MATREP(USPROJER,MEAN(USARCN_USPROJ_ER), 8, 1);
DOCORE USPROJER=MATREP(USPROJER,MEAN(USARCG_USPROJ_ER), 9, 1);
DOCORE USPROJER=MATREP(USPROJER,MEAN(USARRI_USPROJ_ER),10, 1);
DOCORE USPROJER=MATREP(USPROJER,MEAN(USARSB_USPROJ_ER),11, 1);
DOCORE USPROJER=MATREP(USPROJER,MEAN(USAROS_USPROJ_ER),12, 1);
DOCORE USPROJER=MATREP(USPROJER,MEAN(USYDWH_USPROJ_ER), 7, 2);
```

```
DOCORE USPROJER=MATREP(USPROJER,MEAN(USQTDC_USPROJ_ER),18, 16)!
DOCORE USPROJER=MATREP(USPROJER,MEAN(USQTDO_USPROJ_ER),19, 16)!
DO PRREVBF= (USPSBF_USPROJ)*(USQSBF_USPROJ)-(USPSBF_USBASE)*(USQSBF_USBASE) !
DO PRREVPK= (USPSPK_USPROJ)*(USQSPK_USPROJ)-(USPSPK_USBASE)*(USQSPK_USBASE) !
DO PRREVML= (USPSML_USPROJ)*(USQSML_USPROJ)-(USPSML_USBASE)*(USQSML_USBASE) !
DO PRREVDM= (USPSDM_USPROJ)*(USQSDM_USPROJ)-(USPSDM_USBASE)*(USQSDM_USBASE) !

DO RTDPNDC=((((USQTDC_USPROJ)/(USQSDC_USPROJ))-((USQTDC_USBASE)/(USQSDC_USBASE)
))/((USQTDC_USBASE)/(USQSDC_USBASE)))*100.0 !
DO RTDPNDO=((((USQTDO_USPROJ)/(USQSDO_USPROJ))-((USQTDO_USBASE)/(USQSDO_USBASE)
))/((USQTDO_USBASE)/(USQSDO_USBASE)))*100.0 !
DOCORE USPROJPR=MATREP( USPROJPR,MEAN(RTDPNBF), 1,21) !
DOCORE USPROJPR=MATREP( USPROJPR,MEAN(RTDPNPK), 2,21) !
DOCORE USPROJPR=MATREP( USPROJPR,MEAN(RTDPNML), 3,21) !
DOCORE USPROJPR=MATREP( USPROJPR,MEAN(RTDPNPM), 5,21) !
DOCORE USPROJPR=MATREP( USPROJPR,MEAN(RTDPNPE), 6,21) !
DOCORE USPROJPR=MATREP( USPROJPR,MEAN(RTDPNWH), 7,21) !
DOCORE USPROJPR=MATREP( USPROJPR,MEAN(RTDPNCN), 8,21) !
DOCORE USPROJPR=MATREP( USPROJPR,MEAN(RTDPNCG), 9,21) !
DOCORE USPROJPR=MATREP( USPROJPR,MEAN(RTDPNRI),10,21) !
DOCORE USPROJPR=MATREP( USPROJPR,MEAN(RTDPNSB),11,21) !
DOCORE USPROJPR=MATREP( USPROJPR,MEAN(RTDPNOS),12,21) !
DOCORE USPROJPR=MATREP( USPROJPR,MEAN(RTDPNSM),13,21) !
DOCORE USPROJPR=MATREP( USPROJPR,MEAN(RTDPNSO),14,21) !
DOCORE USPROJPR=MATREP( USPROJPR,MEAN(RTDPNOM),15,21) !
DOCORE USPROJPR=MATREP( USPROJPR,MEAN(RTDPNOO),16,21) !
DOCORE USPROJPR=MATREP( USPROJPR,MEAN(RTDPNDB),17,21) !
DOCORE USPROJPR=MATREP( USPROJPR,MEAN(RTDPNDC),18,21) !
DOCORE USPROJPR=MATREP( USPROJPR,MEAN(RTDPNDO),19,21) !
DO USPROJPR=USPROJPR!
DOCORE OPRTMAT(USPROJPR,USQDROWE'L,USPROCOL'L,0,0,-23,USPROJPT'L,1)!
DOCORE PRINT(MEAN(USARTT_USPROJ_PCER))!
DOCORE PRINT(MEAN(USTTRL_USPROJ_PCER))!
DOCORE PRINT(MEAN(USQMDM_USPROJ_PCER))!
DOCORE PRINT(MEAN(USCGAU_USPROJ_PCER))!
DOCORE PRINT(MEAN(USLPI_USPROJ_PCER))!
DOCORE PRINT(MEAN(USOSPM_USPROJ_PCER))!
DOCORE PRINT(MEAN(USSBPM_USPROJ_PCER))!
```

I-B.8. Program to Print Values for Selected Variables

Output data from a model simulation is stored in a TROLL user named DSET. These output DSETS can be displayed and manipulated using TROLL OPRTDSET and other commands as described in the TROLL manuals. This section presents two prepared TROLL programs which can easily provide "readable" TROLL simulation output. They have been set up to efficiently use TROLL's "MACRO" command facility.

I-B.8.a. DISPLAY - Display Selected Variables from an Output DSET.

DISPLAY is a short TROLL MACRO that can be used to print variables that are in a DSET in matrix form. Five parameters are passed to the Macro.

1/ The name of the DSET.
2/ A range of dates (ex. 1975 to 1980) or "ALL".
3/ A heading and/or footnote for the table.
4/ A list of the specific variables to be printed.
5/ A list of column headings (optional).

All parameters should be put in double quotes. The heading lines and the list of variables can be on more than one card. The macro has been tested with a list of 18 variables and works well, however, a longer list could cause an error because of TROLL line length restrictions (200 chr.). If there is a comma in the heading, TROLL assumes that everything following the comma is a footnote. If column headings are not used, then two double quotes should be put in its place.

Example:

```
&DISPLAY "USBASE" "ALL"
"QUANTITY DEMANDED"
"USQDWH USQDCN USQDDM"
"WHEAT, CORN, MILK"
```

EXAMPLE TABLE

QUANTITY DEMANDED

	WHEAT	CORN	MILK
1976	19598.	13636.	20536.7
1977	20495.6	13586.6	27061.3
1978	20745.	13965.	27141.6
1979	19951.3	14103.4	27444.6
1980	20170.1	14012.6	27451.9

Listing of a Longer "DISPLAY" Routine

[redacted].[redacted].DISPLAY.DATA

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
&ERROR &IGNORE
&DISPLAY "LKUSBASE" "1976 TO 1990"
"SUMMARY OUTPUT OF LINKED RUN - USGOL SUPPLY QUANTITY (1000 MT)"
"USQSBF USQSPK USQSML USQSDM USQSPM USQSPE
 USQSWH USQSCN USQSCG USQSRI USQSSB USQSOS
 USQSSM USQSSO USQSOM USQSOO
 USQSDB USQSDC USQSDO"
 "BEEF, PORK, MUTTON+LAMB, DAIRY-MILK, POULTRY-MEAT,
 POULTRY-EGGS, WHEAT, CORN, OTHER COARSE GRAINS, RICE,
 SOYBEANS, OTHER OILSEEDS, SOYMEAL, SOYOIL, OTHER MEALS,
 OTHER OILS, DAIRY-BUTTER, DAIRY-CHEESE, DAIRY-OTHER PRODUCTS"
&DISPLAY "LKUSBASE" "1976 TO 1990"
"SUMMARY OUTPUT OF LINKED RUN - USGOL TRADE QUANTITY (1000 MT)"
"USQTBF USQTPK USQTML  USQTPM USQTPE
 USQTWH USQTCN USQTCG USQTRI USQTSB USQTOS
 USQTSM USQTSO USQTOM USQTOO
 USQTDB USQTDC USQTDO"
 "BEEF, PORK, MUTTON+LAMB , POULTRY-MEAT,
 POULTRY-EGGS, WHEAT, CORN, OTHER COARSE GRAINS, RICE,
 SOYBEANS, OTHER OILSEEDS, SOYMEAL, SOYOIL, OTHER MEALS,
 OTHER OILS, DAIRY-BUTTER, DAIRY-CHEESE, DAIRY-OTHER PRODUCTS"
&DISPLAY "LKUSBASE" "1976 TO 1990"
"SUMMARY OUTPUT OF LINKED RUN - USGOL SUPPLY PRICE (1000 MT)"
"USPSBF USPSPK USPSML USPSDM USPSPM USPSPE
 USPSWH USPSCN USPSCG USPSRI USPSSB USPSOS
 USPSSM USPSSO USPSOM USPSOO
 USPSDB USPSDC USPSDO"
 "BEEF, PORK, MUTTON+LAMB, DAIRY-MILK, POULTRY-MEAT,
 POULTRY-EGGS, WHEAT, CORN, OTHER COARSE GRAINS, RICE,
 SOYBEANS, OTHER OILSEEDS, SOYMEAL, SOYOIL, OTHER MEALS,
 OTHER OILS, DAIRY-BUTTER, DAIRY-CHEESE, DAIRY-OTHER PRODUCTS"
&DISPLAY "LKRWBASE" "1976 TO 1990"
"SUMMARY OUTPUT OF LINKED RUN - RWGOL SUPPLY QUANTITY (1000 MT)"
"RWQSBF RWQSPK RWQSML RWQSPM RWQSPE
 RWQSWH RWQSCN RWQSCG RWQSRI RWQSSB RWQSOS
 RWQSSM RWQSSO RWQSOM RWQSOO
 RWQSDB RWQSDC RWQSDO"
 "BEEF, PORK, MUTTON+LAMB, POULTRY-MEAT,
 POULTRY-EGGS, WHEAT, CORN, OTHER COARSE GRAINS, RICE,
 SOYBEANS, OTHER OILSEEDS, SOYMEAL, SOYOIL, OTHER MEALS,
 OTHER OILS, DAIRY-BUTTER, DAIRY-CHEESE, DAIRY-OTHER PRODUCTS"
&DISPLAY "LKWDBASE" "1976 TO 1990"
"SUMMARY OUTPUT OF LINKED RUN - WDGOL WORLD TRADE PRICE(1000 MT)"
"WDPTBF WDPTPK WDPTML WDPTPM WDPTPE
 WDPTWH WDPTCN WDPTCG WDPTRI WDPTSB WDPTOS
 WDPTSM WDPTSO WDPTOM WDPTOO
 WDPTDB WDPTDC WDPTDO"
 "BEEF, PORK, MUTTON+LAMB, POULTRY-MEAT,
 POULTRY-EGGS, WHEAT, CORN, OTHER COARSE GRAINS, RICE,
 SOYBEANS, OTHER OILSEEDS, SOYMEAL, SOYOIL, OTHER MEALS,
 OTHER OILS, DAIRY-BUTTER, DAIRY-CHEESE, DAIRY-OTHER PRODUCTS"
```

Listing of the Display MACRO Program

[redacted].[redacted].DISPLAY.MACRO.DATA

```
TROLL_MACRO_DISPLAY
&IGNORE
  &1"SIMULATION SUFFIX ?"
  &2"DATES OR ALL ?"
  &3"MAIN HEADING ?"
  &4"VARIABLES ?"
  &5"COLUMN HEADINGS ?"
&END
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7;
CHOMAT LABEL USETS &1,RANGE &2,VARI &4;
LEDIT X; ADD TOP,&3; FILE;
&SETC &6 = "&1_COL" &END
LEDIT Y; ADD TOP,&5; FILE;
&IF "&5" CNE "" &SETC &6 = "Y" &END &IFEND
DO UPRTMAT(DATA_&1,&1_ROW'L,&6'L,0,0,4,X'L,1);
DELETE DATA DATA_&1;
DELETE GENERAL LABEL_X LABEL_Y LABEL_&1_ROW LABEL_&1_COL;
```

I-B.8.b. VARTAB - Tabulate Output for Selected Variables.

The programs that follow are TROLL macros that have been developed for printing GOL data in tabular form. All of these programs have been generalized so that they can print data for any country. There are many programs that work together in printing commodity specific variables. These programs can print most GOL commodity specific variables immediately because headings have been predefined. Other variables can be printed by defining appropriate headings.

There are also four macros that calculate data files. These are:

PC - calculates per capita consumption
RS - calculates real supply prices
PAG - calculates a deflator for GOL commodity prices
RD - calculates real demand prices

Because PC, RS; and RD compute commodity specific variables and create their own headings, other program can be used to print this data in tables.

In addition to the programs used for printing commodity specific variables, there is a macro called DISPLAY which can print any variables that are in a DSET.

The following is a description of four TROLL macros that make it very easy to print simple reports of GOL commodity specific variables. Although any commodity specific variable can be printed using these macros, subheadings exist only for those variables listed below.

Variable Code	Subheading	Units
QD	Quantity Demanded	1000 MT
QS	Quantity Supplied	1000 MT
QT	Quantity Traded	1000 MT
SK	Ending Stocks	1000 MT
EQ	Export Quota	1000 MT
MQ	Import Quota	1000 MT
PE	Price Estimate (Demand)	(*/MT)
MD	Margin (Domestic)	"
MT	Margin (Trade)	"
PD	Price (Demand)	"
PT	Price (Trade)	"
PS	Price (Supply)	"
TC	Tax(+)/Subsidy(-) (Consumption)	"
TE	Tax(+)/Subsidy(-) (Exports)	"
TP	Tax(+)/Subsidy(-) (Production)	"
TM	Tax(+)/Subsidy(-) (Imports)	"

* units of local currency per metric ton.

In order to make the preparation of similar reports as simple as possible the method of specifying report parameters was broken down into two steps. In the first step (The Macro SETRPT) the user provides information that usually does not change over several reports. In the second step (The Macro VARTAB) the user specifies only the variable code (above) for the item he wishes to print.

The parameters needed by the first step (SETRPT) are:

1/ Report destination: ONLINE or OFFLINE
2/ The name of the country: ex. UNITED STATES
3/ The GOL country code: ex. US
4/ An abbreviation for the local currency: ex. US$
5/ The dates to be used in the reports: ex. 1975 TO 1980
6/ Optional: Simulation suffix: ex. USBASE

To pass a parameter to a TROLL macro the user should put each parameter in double quotes after the macro call. It does not matter if the parameters are spread out over many lines.

Example use of SETRPT:

```
&SETRPT "OFFLINE"
"UNITED STATES" "US" "US$"
"1975 TO 1980"
""
```

Notice that when the simulation suffix is omitted, two double quotes are put in its place.

At this point the user can use the macro VARTAB to print any of the variables listed above.

Example use of VARTAB:

```
&VARTAB "QD"
```

If many reports are needed, the user can use the TROLL CALLMAC procedure with VARTAB, instead of a series of calls to VARTAB.

Example:

```
&CALLMAC VARTAB "QD" "QS";
```

The resulting reports are shown in examples 1 and 2. If the user wants to change any of the information he can re-execute the SETRPT macro.

To print reports of items other than those listed above , the user need only define the subheading. The subheading is created using the TEXT command of the REPORT WRITER. The first six characters of the subheading name must be CRPTSH and the last two must be the variable code.

Example: Create a subheading for per capita consumption data which is the item PC.

```
REPORT;
TEXT CRPTSHPC C UNDERSCORE - WID 132 132:
<PER CAPITA CONSUMPTION>, <(KILOGRAMS PER PERSON)>;
QUIT;
```

To clear all report specifications that are put on the users TROLL file the user should execute the macro CLRPT.

Example:

```
&CLRPT
```

The macro SETPERM is used only when the TROLL file containing these macros is initialized. Users will not need to use it.

Listing 1 is an example of a batch job that uses these macros. File FT10F001 is the users file and File FT12F001 contains the macros.

Example 1

HISTORICAL DATA - UNITED STATES - LIVESTOCK AND PRODUCTS

QUANTITY DEMANDED

(1000 MT)

	BEEF AND VEAL	PORK	MUTTON AND LAMB	POULTRY-MEAT	POULTRY-EGGS	DAIRY-MILK	DAIRY-CHEESE	DAIRY-BUTTER	DAIRY-OTHER PRODUCTS
1975	12080	5473	197	4813	3813	26533	1398	462	318
1976	13029	5843	181	5181	3801	26542	1634	427	353
1977	12525	6486	169	5369	3803	26488	1697	429	342
1978	12224	6146	156	5700	3926	26567	1700	440	338
1979	10974	7085	153	6230	4049	26536	1751	459	369
1980	10866	7640	163	6386	4020	26359	1793	460	354

HISTORICAL DATA - UNITED STATES - CROPS AND PRODUCTS

QUANTITY DEMANDED

(1000 MT)

	WHEAT	CORN	OTHER COARSE GRAINS	RICE	SOYBEANS	OTHERDS OILSEEDS	SOYMEAL	OTHER MEALS	SOYOIL	OTHER OILS
1975	18697	12446	5921	1385	2093	1173	1	116	2986	750
1976	18508	13030	5074	1618	1942	1683	1	121	3612	872
1977	18125	13995	5098	1248	2078	1682	1	114	3406	796
1978	18449	14605	5291	1686	2231	1949	1	110	3752	906
1979	18942	17168	5426	1817	2691	1643	1	113	4056	852
1980	19595	19051	5347	1781	2305	2103	1	211	4073	922

Example 2

HISTORICAL DATA - UNITED STATES - LIVESTOCK AND PRODUCTS

QUANTITY SUPPLIED

(1000 MT)

	BEEF AND VEAL	PORK	MUTTON AND LAMB	POULTRY-MEAT	POULTRY-EGGS	DAIRY-MILK	DAIRY-CHEESE	DAIRY-BUTTER	DAIRY-OTHER PRODUCTS
1975	11271	5343	186	4845	3825	52344	1275	445	451
1976	12166	5755	168	5385	3821	54513	1506	444	420
1977	11845	6009	159	5535	3843	55635	1523	493	502
1978	11283	6075	140	5880	3984	55094	1597	451	417
1979	9925	7008	133	6507	4500	55978	1585	447	412
1980	10002	7535	146	6647	4101	58253	1790	518	522

HISTORICAL DATA - UNITED STATES - CROPS AND PRODUCTS

QUANTITY SUPPLIED

(1000 MT)

	WHEAT	CORN	OTHER COARSE GRAINS	RICE	SOYBEANS	OTHERDS OILSEEDS	SOYMEAL	OTHER MEALS	SOYOIL	OTHER OILS
1975	57883	148357	37086	4050	33098	6387	15268	2223	3435	800
1976	59474	159750	34614	3781	42138	5605	18836	1907	4238	687
1977	55606	165163	40557	3121	34986	6079	17205	1992	3871	717
1978	48245	184551	37508	4276	48210	8367	20176	2324	4540	837
1979	58174	201687	37101	4331	50755	7688	22161	2303	4986	829
1980	64557	168919	29842	4772	61683	10788	24458	2482	5503	894

Listing 1

[redacted].[redacted].REPORT.CNTL

```
//[redacted] JOB ([redacted],RJ013),'TROLL REPORT',CLASS=C,
//  MSGLEVEL=1,TIME=(,20),PRTY=3
/*ROUTE PRINT RMT13
//STEP1 EXEC ERSTROLL
//FT05F001 DD *
MATCH
SEARCH [redacted];
&SETRPT "OFFLINE" "UNITED STATES" "US" "USS" "1975 TO 1980" ""
&CALLMAC VARTAB "QD" "QS";
&CLRPT
LOGOUT;
/*
//FT10F001 DD DSN=[redacted].[redacted].GOL.TROLL.DATA,
// UNIT=SYSDA,DISP=OLD
//FT14F001 DD DSN=[redacted].[redacted].REPORT.DATA,
// UNIT=SYSDA,DISP=OLD
!!
```

The Macro SETRPT puts the parameters passed to it in global variables (CIFARGS) so that they can be accessed by VARTAB. It also creates additional CIFARGS. The contents of the CIFARGs are explained below:

CIFARG NUMBER	CONTENTS
1	Report Destination: ONLINE or OFFLINE
2	Country Name
3	Country Code
4	Local Currency
5	Dates
6	Simulation Suffix (preceded by an underscore)
7	"SIMULATION OUTPUT" or "HISTORICAL DATA"
8	Number of characters in country name.
9	Name of FORMS to be used default is "CRPTF.0"

The CIFARGS used by the report programs (SETRPT, VARTAB, etc.) can be altered using the macro CHNAGE. One use of this is to charge the number of significant digits printed in the report. CIFARG (9) contains the name of the forms statement used in printing. CIFARG (9) normally contains "CRPTF.0" which prints integers only. To print one significant digit you can change CIFARG (9) to "CRPTF.1". Up to four significant digits ("CRPTF.4") can be printed in the tables.

The general format of the statement is: &CHANGE "CIFARG NUMBER" "TEXT TO INSERT"

Example: Change CIFARG (9) to "CRPTF.2"

```
&CHANGE "9" "CRPTF.2"
```

The following is a description of four TROLL macros that create data files:

Macro Name	Computed Information	Data File Created
PC	Per capita consumption	cc PC **
RS	Real supply prices	cc RS **
PAG	Deflator for agricultural products	cc PAG
RD	Real demand prices	cc RD **

cc is a two digit country code.
** indicates that the data is created for each commodity.

All of these programs reference CIGARGS that are initialized by the program SETRPT. This makes the programs independent of country codes and dates.

The data files created can be permanent or temporary. To create a temporary data file, the macro should be called like this:

```
&PC  "SAVE"  or &PC  ""
```

To create permanent data files call the macro like this:

```
&PC  "FILE"
```

The data created by the program PAG (a deflator used in calculating real demand prices) is created for all years available and it is also printed. The programs PC, RD, and RS create data only for the range of years given to SETRPT. This data is not automatically printed out. To print tables of this data you can use the macro VARTAB.

Example 1:

Compute and file per capita consumption data for the United States for the years 1975 to 1980. After computing this information, print it in a table.

```
&SETRPT "OFFLINE" "UNITED STATES" "US" "US$"
"1975 to 1980" ""

&PC "FILE"
&VARTAB "PC"
```

Example 2:

Compute real demand prices for CANADA 1975 to 1980, and print a table.

```
&SETRPT "OFFLINE" "CANADA" "CN" "CN$"
"1975 to 1980"

&PAG "SAVE"
&RD "FILE"
&VARTAB "RD"
```

Note: In order to calculate real demand prices you must create the constant ccWTNG in your model. This constant represents the percentage of disposable income/expenditures spent on food.

Example: CEDIT USGOL; ADD USWTNG 0.16; FILE;

[redacted].[redacted].PL.MACRO.DATA

```
TROLL_MACRO_PL
&IGNORE &1"FILE OR NULL" &END
&SETC &3 = "SAVE" &END
&IF &1 LEQ "FILE" &SETC &3 = "" &END &IFEND
&SETC &1 = "&CIFARG(3)" &END
&SETC &2 = "&CIFARG(6)" &END
DORANGE &CIFARG(5);
DO&3 &1PCBF&2 = &1QDBF&2 / &1POP&2;
DO&3 &1PCPK&2 = &1QDPK&2 / &1POP&2;
DO&3 &1PCML&2 = &1QDML&2 / &1POP&2;
DO&3 &1PCDM&2 = &1QDDM&2 / &1POP&2;
DO&3 &1PCPM&2 = &1QDPM&2 / &1POP&2;
DO&3 &1PCPE&2 = &1QDPE&2 / &1POP&2;
DO&3 &1PCWH&2 = &1QDWH&2 / &1POP&2;
DO&3 &1PCCN&2 = &1QDCN&2 / &1POP&2;
DO&3 &1PCCG&2 = &1QDCG&2 / &1POP&2;
DO&3 &1PCRI&2 = &1QDRI&2 / &1POP&2;
DO&3 &1PCSB&2 = &1QDSB&2 / &1POP&2;
DO&3 &1PCDS&2 = &1QDDS&2 / &1POP&2;
DO&3 &1PCSM&2 = &1QDSM&2 / &1POP&2;
DO&3 &1PCSO&2 = &1QDSO&2 / &1POP&2;
DO&3 &1PCDM&2 = &1QDDM&2 / &1POP&2;
DO&3 &1PCDD&2 = &1QDDD&2 / &1POP&2;
DO&3 &1PCDP&2 = &1QDDP&2 / &1POP&2;
DO&3 &1PCDC&2 = &1QDDC&2 / &1POP&2;
DO&3 &1PCDD&2 = &1QDDD&2 / &1POP&2;
REPORT;
TEXT CRPTSHPC C UNDERSCORE = WID 132 132;
<PER CAPITA CONSUMPTION>,<(KILOGRAMS PER PERSON)>;
QUIT;
```

Listing of TROLL MACROs

[redacted].[redacted].RS.MACRO.DATA

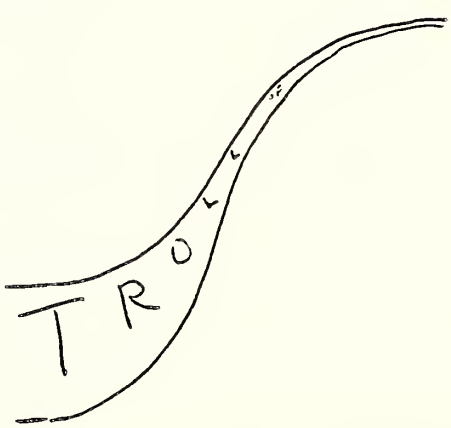
```
TROLL_MACRO_RS
&IGNORE &1"FILE OR NULL" &END
&SETC &3 = "SAVE" &END
&IF &1 EQ "FILE" &SETC &3 = "" &END &IFEND
&SETC &1 = "&CIFARG(3)" &END
&SETC &2 = "&CIFARG(6)" &END
DORANGE &CIFARG(5);
DO&3 &1RSBF&2=((&1PSBF&2 * 100)/&1ICP&2);
DO&3 &1RSPK&2=((&1PSPK&2 * 100)/&1ICP&2);
DO&3 &1RSML&2=((&1PSML&2 * 100)/&1ICP&2);
DO&3 &1RSDM&2=((&1PSDM&2 * 100)/&1ICP&2);
DO&3 &1RSPM&2=((&1PSPM&2 * 100)/&1ICP&2);
DO&3 &1RSPE&2=((&1PSPE&2 * 100)/&1ICP&2);
DO&3 &1RSWH&2=((&1PSWH&2 * 100)/&1ICP&2);
DO&3 &1RSCN&2=((&1PSCN&2 * 100)/&1ICP&2);
DO&3 &1RSCG&2=((&1PSCG&2 * 100)/&1ICP&2);
DO&3 &1RSRI&2=((&1PSRI&2 * 100)/&1ICP&2);
DO&3 &1RSSB&2=((&1PSSB&2 * 100)/&1ICP&2);
DO&3 &1RSDS&2=((&1PSDS&2 * 100)/&1ICP&2);
DO&3 &1RSSM&2=((&1PSSM&2 * 100)/&1ICP&2);
DO&3 &1RSSO&2=((&1PSSO&2 * 100)/&1ICP&2);
DO&3 &1RSDM&2=((&1PSDM&2 * 100)/&1ICP&2);
DO&3 &1RSDD&2=((&1PSDD&2 * 100)/&1ICP&2);
DO&3 &1RSDB&2=((&1PSDB&2 * 100)/&1ICP&2);
DO&3 &1RSDC&2=((&1PSDC&2 * 100)/&1ICP&2);
DO&3 &1RSDD&2=((&1PSDD&2 * 100)/&1ICP&2);
REPORT;
TEXT CRPTSHRS C UNDERSCORE = WID 132 132;
<REAL SUPPLY PRICES>,<(1970 &CIFARG(4) PER METRIC TON)>;
QUIT;
```

[redacted].[redacted].PAG.MACRO.DATA

```
TROLL_MACRO_PAG
&IGNORE &1"FILE OR NULL" &END
&SETC &3 = "SAVE" &END
&IF &1 CFG "FILE" &SETC X3 = "" &END &IFEND
&SETC X1 = "&CIFARG(3)" &END
&SETC &2 = "&CIFARG(6)" &END
DURANGE;
DUCURE &1PAG&2 =
(VALUE(&1QDBF&2,1976) * &1PDBF&2) +
(VALUE(&1QDPK&2,1976) * &1PDPK&2) +
(VALUE(&1QDML&2,1976) * &1PDML&2) +
(VALUE(&1QDDM&2,1976) * &1PDDM&2) +
(VALUE(&1QDPM&2,1976) * &1PDPM&2) +
(VALUE(&1QDPE&2,1976) * &1PDPE&2) +
(VALUE(&1QDWH&2,1976) * &1PDWH&2) +
(VALUE(&1QDCN&2,1976) * &1PDCN&2) +
(VALUE(&1QDCG&2,1976) * &1PDCG&2) +
(VALUE(&1QDRI&2,1976) * &1PDRI&2) +
(VALUE(&1QDSB&2,1976) * &1PDSB&2) +
(VALUE(&1QDDS&2,1976) * &1PDDS&2) +
(VALUE(&1QDSM&2,1976) * &1PDSM&2) +
(VALUE(&1QDSU&2,1976) * &1PDSU&2) +
(VALUE(&1QDDM&2,1976) * &1PDDM&2) +
(VALUE(&1QDDU&2,1976) * &1PDDU&2) +
(VALUE(&1QDDB&2,1976) * &1PDDB&2) +
(VALUE(&1QDDC&2,1976) * &1PDDC&2) +
(VALUE(&1QDDO&2,1976) * &1PDDO&2);
DUCURE &1PAG&2 = &1PAG&2/
(VALUE(&1QDBF&2,1976) +
VALUE(&1QDPK&2,1976) +
VALUE(&1QDML&2,1976) +
VALUE(&1QDDM&2,1976) +
VALUE(&1QDPM&2,1976) +
VALUE(&1QDPE&2,1976) +
VALUE(&1QDWH&2,1976) +
VALUE(&1QDCN&2,1976) +
VALUE(&1QDCG&2,1976) +
VALUE(&1QDRI&2,1976) +
VALUE(&1QDSB&2,1976) +
VALUE(&1QDDS&2,1976) +
VALUE(&1QDSM&2,1976) +
VALUE(&1QDSU&2,1976) +
VALUE(&1QDDM&2,1976) +
VALUE(&1QDDU&2,1976) +
VALUE(&1QDDB&2,1976) +
VALUE(&1QDDC&2,1976) +
VALUE(&1QDDO&2,1976));
DU&3 &1PAG&2 = &1PAG&2 * 100 / VALUE(&1PAG&2,1976);
OPRTDATA ALL &1PAG&2;
```

[redacted].[redacted].RD.MACRO.DATA

```
TROLL_MACRO_RD
&IGNORE &1"FILE OR NULL" &END
&SETC &3 = "SAVE" &END
&IF &1 CEQ "FILE" &SETC &3 = "" &END &IFEND
&SETC &1 = "&CIFARG(3)" &END
&SETC &2 = "&CIFARG(6)" &END
DOPARGE &CIFARG(5);
USEMOD &1GDL;
DO&3 &1PNA&2 = (&1WTNG'C * &1PNG&2)+((1-&1WTNG'L)*&1PAG&2);
DO&3 &1RDBF&2 = (&1PDBF&2*100)/&1PNA&2;
DO&3 &1RDPK&2 = (&1PDPK&2*100)/&1PNA&2;
DO&3 &1RDML&2 = (&1PDML&2*100)/&1PNA&2;
DO&3 &1RDDM&2 = (&1PDDM&2*100)/&1PNA&2;
DO&3 &1RDPM&2 = (&1PDPM&2*100)/&1PNA&2;
DO&3 &1RDPE&2 = (&1PDPE&2*100)/&1PNA&2;
DO&3 &1RDWH&2 = (&1PDWH&2*100)/&1PNA&2;
DO&3 &1RDCN&2 = (&1PDCN&2*100)/&1PNA&2;
DO&3 &1RDLG&2 = (&1PDLG&2*100)/&1PNA&2;
DO&3 &1RDRI&2 = (&1PDRI&2*100)/&1PNA&2;
DO&3 &1RDSH&2 = (&1PDSH&2*100)/&1PNA&2;
DO&3 &1RDUS&2 = (&1PDUS&2*100)/&1PNA&2;
DO&3 &1RDSM&2 = (&1PDSM&2*100)/&1PNA&2;
DO&3 &1RDSO&2 = (&1PDSO&2*100)/&1PNA&2;
DO&3 &1RDOM&2 = (&1PDOM&2*100)/&1PNA&2;
DO&3 &1RDOO&2 = (&1PDOO&2*100)/&1PNA&2;
DO&3 &1RDDB&2 = (&1PDDB&2*100)/&1PNA&2;
DO&3 &1RDDC&2 = (&1PDDC&2*100)/&1PNA&2;
DO&3 &1RDDO&2 = (&1PDDO&2*100)/&1PNA&2;
REPORT;
TEXT LRPTSHRD C UNDERSCORE = WID 132 132;
«REAL DEMAND PRICES»,«(1976 &CIFARG(4) PER METRIC TON)»;
QUIT;
```



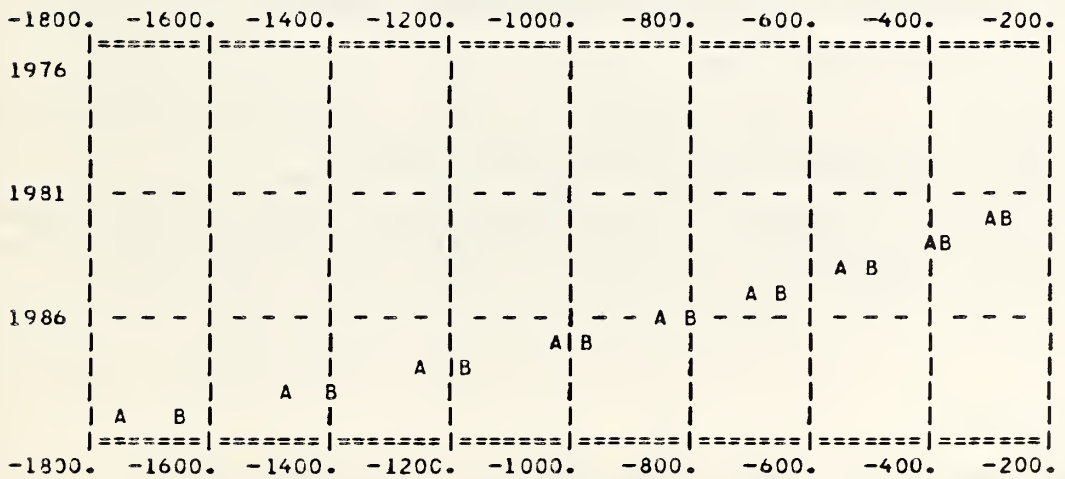
I-B.9. Plotting Simulation Output - OPLTDSET

The output variables of a simulation run can be easily plotted using the TROLL OPLTDSET command. A sample set of commands along with a sample plot of a variable are presented below.

```
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7 FULEGEND ;
RTITLE GCL LINKED SIMULATION RUN ;
OPLTDSET VARIABLES
USQSBF USQSPK USQSML USQSDM USQSPM USQSPE
USQSWH USQSCN USQSCG USQSRI USQSSB USQSOS
USQSSM USQSSO USQSQM USQSOO
USQSDB USQSDC USQSDO ,
RANGE 1976 TO 1990,
DSETS USGOL LKUSBASE;
OPLTDSET VARIABLES
USQTBF USQTPK USQTML USQTPM USQTPE
USQTWH USQTCN USQTCG USQTRI USQTSB USQTOS
USQTSM USQTSO USQTQM USQTOO
USQTDB USQTDC USQTDO ,
RANGE 1976 TO 1990,
DSETS USGOL LKUSBASE;
```

SIMULATION OUTPUT BY VARIABLE

USQTPE - ENDOGENOUS



**********LEGEND**********

TIME BOUNDS: 1976 TO 1990

SYMBOL	SCALE	NAME
A	#1	USGOL
B	#1	LKUSBASE

****

II. Computer Routines for Creating and Running Programs of TROLL Commands for GOL Models

GOL model components can be created, managed, and run using sets of TROLL commands for various modeling tasks. Given the repetitive nature of the standard models and given the desire to repeat similar TROLL commands for different commodities and countries, it is efficient to have a computer generate as many of the repetitive commands as possible. This saves editing and computer connect time.

Two types of computer routines are presented below. The first type consists of sets of TSO programs and commands which operate on the WCC computer. 4/ The second type consists of BASIC programs which operate on a CP/M micro computer system and generate TROLL commands or programs which can be transmitted from the micro computer to the mainframe computer which contains the TROLL package. 5/ Also, a few utility programs are presented which may prove helpful to a researcher modeling in TROLL using the tools outlined in this report.

II-A. Mainframe and Utility Programs for Building TROLL Programs and Operating GOL Models.

The construction, operation, and general maintenance and improvement of a GOL country model is most easily done by using a set of utility programs of TSO commands which accomplish specific tasks. In addition, job control programs are needed to run TROLL programs. This section documents these utility programs and job control commands.

II-A.1. Utility Programs for Building TROLL Programs

A set of utility programs of TSO commands can be used to create a new standard model and supporting programs and obtain printed or punched copies of such programs.

4/ For details about the Time Sharing Option (TSO) see; OS/VS2 TSO Command Language Reference, IBM Corporations, Poughkeepsie, N.Y., 1978.

5/ For details about the CP/M (trademark) operating system for micro computers see: An Introduction to CP/M Features and Facilities, Digital Research Corp., Pacific Grove, California, January 1978. The BASIC programs presented in this report are written in Microsoft BASIC (trademark) but would run in other versions with some modification. For details on Microsoft BASIC, see the appropriate manuals which come with the version installed on the micro computer in use.

II-A.1.a. CREATE.CLIST - Clone a New Country Model from USGOL

'[redacted].[redacted].CREATE.CLIST' uses USGOL programs to create similar programs for other countries. It does this by changing all occurances of "UNITED STATES" to the new country name, "US$" to the new currency name, and "US" to the new country code. If any changes are made to the words "EXOGENOUS", "ENDOGENOUS", "CUSTOM", "CRUSH", or "USEMOD" then these changes are corrected. Any other words containing the characters "US" will be changed, therefore these words must be corrected by the user.

In the following example the file '[redacted].[redacted].USGOLSMC.DATA' is used to create a similar file for the European Community. The new file that is created is called '[redacted].[redacted].ECGOLSMC.DATA'. Notice in the example that when the user is asked for the GOL file code, only three (3) characters need be entered to identify the file (SMC).

The first prompt is for the country code, this code is by definition restricted to two characters. The second prompt is for the country name, this is limited to two words (1 imbedded space) by the CLIST. The third prompt is for the currency code, this is limited to one word by the CLIST. After the third prompt the information given by the user is echoed back so the user can verify that it is correct. If any corrections need to be made, then the user can enter the corresponding word in parenthesis and the program will prompt for the correct information. When all corrections have been made, the user should press carriage return. Once this information is entered, the user can 'create' as many programs as needed.

The next prompt is for the GOL file name. After this is entered, the user must verify it. If it is incorrect, then the user can enter "N" if it is correct the user can enter "Y" or just press carriage return. The program will then print out some messages that can be ignored. After the program has created the new file, the user is asked whether it should be printed offline or put on punch cards. If the user enters only PUNCH, then the file will be put on punch cards and printed offline (same as entering PRINT PUNCH). After this the user is again asked for a GOL file to 'create' if the user wants to exit he should press carriage return at this prompt.

SAMPLE RUN

```
exec [redacted].create.clist

ENTER COUNTRY CODE FOR FILE TO BE CREATED

ec

ENTER THE NAME OF THE COUNTRY (EC)

european community

ENTER TITLE OF NEW CURRENCY (EC)

ecu

THE INPUT YOU HAVE GIVEN IS:
 COUNTRY CODE--------(CODE)------EC
 COUNTRY NAME--------(NAME)------EUROPEAN COMMUNITY
 CURRENCY------------(CURR)------ECU
ENTER "SHOW" TO REDISPLAY INPUT

ENTER THE NAME OF THE ITEM TO CHANGE OR CARRIAGE RETURN


ENTER GOL FILE NAME (3 CHARS)  (PRESS CARRIAGE RETURN TO EXIT)

smc

VERIFY FILE CODE IS SMC (Y OR N)


DATA SET [redacted].ECGOLSMC.DATA NOT LINE NUMBERED, USING NONUM
TOP OF DATA SET

ENTER "PRINT" AND/OR "PUNCH" (OR CARRIAGE RETURN)

print punch
JOB [redacted](JOB04159) SUBMITTED  **  FREE ALL FILES  **
$HASP165 JOB 4159  [redacted] ENDED CN(00)
ESCS DATA SERVICES CENTER PRINT UTILITY R3:2
PRINT R3:2
DATASET - [redacted].[redacted].ECGOLSMC.DATA
NUMBER OF LINES PRINTED = 1138
NUMBER OF PAGES PRINTED =   23
SYSOUT(A) IS LABELLED [redacted] AND ROUTED TO RMT13

ENTER GOL FILE NAME (3 CHARS)  (PRESS CARRIAGE RETURN TO EXIT)


READY
logoff
***  [redacted]     CONNECT CHARGE -- $        .22                    ***MVS
***  [redacted]     SESSION CHARGE -- $       2.64      APPROX TSO SESSION COST  WITHOUT
 EXPONENTIATION
[redacted] LOGGED OFF TSO AT 16:56:08 ON MAY 25, 1982
LAST STEP COMPLETION CODE WAS USER   000
```

LISTING OF CREATE PROGRAM

```
0010 CONTROL PROMPT NOFLUSH
0020 SET MODE = INPUT
0030 WRITE
0040 PCODE: WRITE
0050 WRITE ENTER COUNTRY CODE FOR FILE TO BE CREATED
0060 WRITE
0070 READ &CODE
0080 IF &LENGTH(&STR(&CODE)) NE 2 THEN +
0090 DO
0100 WRITE COUNTRY CODE MUST BE 2 CHARACTERS LONG.
0110 WRITE
0120 GOTO PCODE
0130 END
0140 IF &MODE EQ EDIT THEN GOTO CHANGE
0150 PNAME: WRITE
0160 WRITE ENTER THE NAME OF THE COUNTRY (&CODE)
0170 WRITE
0180 READ A B C
0181 IF &STR(&C) NE &STR() THEN +
0182 DO
0183   WRITE THE COUNTRY NAME IS LIMITTED TO 2 WORDS PLEASE
0184   WRITE ENTER ANOTHER NAME AT PROMPT
0184   GOTO PNAME
0185 END
0190 SET NAME = &STR(&A &B)
0200 IF &STR(&A) EQ &STR() THEN GOTO PNAME
0210 IF &MODE EQ EDIT THEN GOTO CHANGE
0220 PCURR: WRITE
0230 WRITE ENTER TITLE OF NEW CURRENCY (&CODE)
0240 WRITE
0250 READ A B
0251 IF &STR(&B) NE &STR() THEN +
0252 DO
0253   WRITE CURRENCY NAME IS LIMITTED TO ONE WORD
0254   WRITE PLEASE ENTER NEW CURRENCY NAME AT PROMPT
0255   GOTO PCURR
0256 END
0260 SET CURR = &STR(&A)
0270 IF &STR(&A) EQ &STR() THEN GOTO PCURR
0280 IF &MODE EQ EDIT THEN GOTO CHANGE
0290 SET MODE = EDIT
0300 ERROR GOTO CHGERR
0310 PSHOW: WRITE
0320 WRITE THE INPUT YOU HAVE GIVEN IS:
0330 WRITE  COUNTRY CODE--------(CODE)------&CODE
0340 WRITE  COUNTRY NAME--------(NAME)------&NAME
0350 WRITE  CURRENCY------------(CURR)------&CURR
0360 WRITE ENTER "SHOW" TO REDISPLAY INPUT
0370 CHANGE: WRITE
0380 WRITE ENTER THE NAME OF THE ITEM TO CHANGE OR CARRIAGE RETURN
0390 WRITE
0400 READ ITEM
0410 IF &ITEM NE &STR() THEN GOTO P&ITEM
0420 ERROR OFF
0430 PFILE:WRITE
0440 WRITE ENTER GUL FILE NAME (3 CHARS)  (PRESS CARRIAGE RETURN TO EXIT)
0450 WRITE
```

```
0460 READ FILE
0470 IF &FILE EQ &STR() THEN EXIT
0480 IF &LENGTH(&STR(&FILE)) NE 3 THEN +
0490 DO
0500 WRITE GOL FILE CODE MUST BE 3 CHARACTERS LONG.
0510 GOTO PFILE
0520 END
0530 WRITE
0540 WRITE VERIFY FILE CODE IS &STR(&FILE) (Y' OR N)
0550 WRITE
0560 READ VER
0570 IF &VER EQ N OR &VER EQ NO THEN GOTO PFILE
0580 SET FILE1 = [redacted].USGOL&FILE..DATA
0590 SET FILE2 = [redacted].&CODE.GOL&FILE..DATA
0600 COPY &FILE1 &FILE2
0650 EDIT &FILE2
0660 TOP
0670 * US$ US UNITED STATES  (DUMMY TEXT STRING)
0680 C * 999999 /US$/&CURR/ALL
0690 TOP
0700 C * 999999 /US/&CODE/ALL
0710 TOP
0720 C * 999999 /UNITED STATES/&NAME/ALL
0730 TOP
0740 * EXOGENO&CODE CR&CODE.H ENDOGENO&CODE &CODE.EMOD C&CODE.TOM &CODE.UALLY
0750 C * 999999 /EXOGENO&CODE/EXOGENOUS/ALL
0760 TOP
0761 C * 999999 /&CODE.UALLY/USUALLY/ALL
0762 TOP
0770 C * 999999 /CR&CODE.H/CRUSH/ALL
0780 TOP
0790 C * 999999 /ENDOGENO&CODE/ENDOGENOUS/ALL
0800 TOP
0810 C * 999999 /&CODE.EMOD/USEMOD/ALL
0820 TOP
0830 C * 999999 /C&CODE.TOM/CUSTOM/ALL
0840 TOP
0850 DEL *
0860 END SAVE
0870 WRITE
0880 WRITE ENTER "PRINT" AND/OR "PUNCH" (OR CARRIAGE RETURN)
0890 WRITE
0900 READ OP1 OP2
0910 IF &OP1 EQ &STR() THEN GOTO PFILE
0930 IF &OP1 EQ PUNCH OR &OP2 EQ PUNCH THEN +
0931 DO
0940 PUNCH &SYSPREF..&FILE2 [redacted] [redacted] &SYSUID
0941 PRINT2 &SYSPREF..&FILE2
0941 GOTO PFILE
0941 END
0942 IF &OP1 EQ PRINT OR &OP2 EQ PRINT THEN PRINT2 &SYSPREF..&FILE2
0950 GOTO PFILE
0960 CHGERR:WRITE
0970 WRITE ENTER THE NAME THAT IS IN PARENTHESIS THAT
0980 WRITE CORRESPONDS TO THE INPUT YOU WANT TO CHANGE
0990 WRITE IF YOU DON'T WANT TO MAKE ANY CHANGES THEN
1000 WRITE PRESS CARRIAGE RETURN.
1010 WRITE
1020 GOTO CHANGE
```

II-A.1.b. COPY.CLIST - Build an Equation Set for all GOL Commodities from a Master Equation

The CLIST '[redacted].COPY.CLIST' takes a generalized TROLL statement and from that creates nineteen copies of the statement each specific to one of the 19 commodities used in GOL.

In the generalized statement the user puts the characters '$$' wherever commodity codes are to be put. The generalized statement can be any number of lines long and must be in a TSO dataset. The CLIST assumes that the new statements are to be put on punch cards, however this assumption is easily overridden.

EXAMPLE: In this example the user creates a dataset '[redacted].QTEQ.DATA' that contains the generalized statement. Then the user executes the CLIST and gives it the name of his dataset. The CLIST creates the new statements and puts them in a file '[redacted].C19.QTEQ.DATA'. The new statements are then put on punch cards.

1.

```
EDIT [redacted].QTEQ.DATA
DATA SET OR MEMBER NOT FOUND, ASSUMED TO BE NEW
INPUT
00010 USQT$$: USQT$$'DEF == USQS$$'N-USQD$$'N-USQA$$'POLN;
00020
EDIT
END SAVE
```

Here the user enters the generalized statement in a dataset.

2. exec.'[redacted].copy.clist' '[redacted].qteq.data'

This executes the CLIST and gives it the name of the dataset containing the statement.

The dataset [redacted].C19.QTEQ.DATA is created and put on punch cards. Listing 1 is a printout of [redacted].C19.QTEQ.DATA.

Listing 1

```
USQTBF: USQTBF'DEF == USQSBF'N-USQDBF'N-USQABF'PCLN;
USQTPK: USQTPK'DEF == USQSPK'N-USQDPK'N-USQAPK'PCLN;
USQTML: USQTML'DEF == USQSML'N-USQDML'N-USQAML'PCLN;
USQTPM: USQTPM'DEF == USQSPM'N-USQDPM'N-USQAPM'PCLN;
USQTPE: USQTPE'DEF == USQSPE'N-USQDPE'N-USQAPE'PCLN;
USQTDM: USQTDM'DEF == USQSDM'N-USQDDM'N-USQADM'PCLN;
USQTDC: USQTDC'DEF == USQSDC'N-USQDDC'N-USQADC'PCLN;
USQTDB: USQTDB'DEF == USQSDB'N-USQDDB'N-USQADB'PCLN;
USQTDO: USQTDO'DEF == USQSDO'N-USQDDO'N-USQADO'PCLN;
USQTWH: USQTWH'DEF == USQSWH'N-USQDWH'N-USQAWH'PCLN;
USQTCN: USQTCN'DEF == USQSCN'N-USQDCN'N-USQACN'PCLN;
USQTCG: USQTCG'DEF == USQSCG'N-USQDCG'N-USQACG'PCLN;
USQTRI: USQTRI'DEF == USQSRI'N-USQDRI'N-USQARI'PCLN;
USQTSB: USQTSB'DEF == USQSSB'N-USQDSB'N-USQASB'PCLN;
USQTOS: USQTOS'DEF == USQSOS'N-USQDOS'N-USQAOS'PCLN;
USQTSM: USQTSM'DEF == USQSSM'N-USQDSM'N-USQASM'PCLN;
USQTOM: USQTOM'DEF == USQSOM'N-USQDOM'N-USQAOM'PCLN;
USQTSO: USQTSO'DEF == USQSSO'N-USQDSO'N-USQASO'PCLN;
USQTOO: USQTOO'DEF == USQSOO'N-USQDOO'N-USQAOO'PCLN;
END OF DATA
READY
```

If the user did not want the new statements on punch cards he could have entered:

```
exec '[redacted].[redacted].copy.clist' '[redacted].qteq.data punch(no)'
```

If the user had used the characters '!!' instead of '$$' he could have entered:

```
exec '[redacted].[redacted].copy.clist' '[redacted].qteq.data chgstr(!!)'
```

Listing 2 is a printout of the CLIST.

[redacted].[redacted].COPY.CLIST

```
010 PROC 1 DSNAME,PUNCH(Y) CHGSTR(SS)
020 CONTROL NOFLUSH
030 SET C = &LENGTH(&DSNAME)
040 SET C1 = &SUBSTR(1:5,&DSNAME)
050 SET C2 = &SUBSTR(7:&C,&DSNAME)
060 SET JOB = [redacted]&SUBSTR(4:5,&DSNAME)
070 SET NEWFILE = &STR(&C1..C19.&C2)
080 COPY &DSNAME &NEWFILE RENUM(10 10)
090 SET &CCNT = 19
100 SET CHG0  = AF
110 SET CHG1  = PK
120 SET CHG2  = ML
130 SET CHG3  = DM
140 SET CHG4  = PM
150 SET CHG5  = PE
160 SET CHG6  = WM
170 SET CHG7  = CN
180 SET CHG8  = CG
190 SET CHG9  = RI
200 SET CHG10 = SB
210 SET CHG11 = DS
220 SET CHG12 = SM
230 SET CHG13 = SO
240 SET CHG14 = OM
250 SET CHG15 = DO
260 SET CHG16 = DB
270 SET CHG17 = DC
280 SET CHG18 = DD
290 ALLOC FI(SOURCE) DA(&NEWFILE) SHR
300 SET &SCNT = 0
310 OPENFILE SOURCE INPUT
320 ERROR GOTO DONESL
330 SL:GETFILE SOURCE
340 SET &SCNT = &SCNT + 1
350 GOTO SL
360 DONESL: CLOSFILE SOURCE
370 ERROR OFF
371 FREEALL
380 SET X = 10 * &SCNT + 10
390 SET Y = &X-20
400 SET LCNT = 0
410 SET Z = &SCNT*10
420 SET OLD = &STR(&CHGSTR)
430 EDIT &NEWFILE
440 LEDIT: SET T1 = &X+&Z*&LCNT
450 SET T2 = &T1+&Y
460 SET &NEW = &STR(&&CHG&LCNT)
470 COPY 10 &Z &T1
480 C &T1 &T2 /&OLD/&NEW/ALL
490 SET LCNT = &LCNT + 1
500 IF &LCNT LT &CCNT THEN GOTO LEDIT
510 DELETE 10 &Z
520 UNNUM
530 END SAVE
540 PRTO &NEWFILE CLASS(A) DEST(RMT13)
550 IF &PUNCH EQ Y THEN PUNCH &SYSPREF..&NEWFILE &JOB [redacted] &SYSUID
```

Listing 2

II-A.1.c. PUNCH.CLIST - Punch a TSO File

This CLIST produces a punched card copy of an TSO file with a .DATA suffix. The user is prompted for the name of the file.

[redacted].[redacted].PUNCH.CLIST

```
010  WRITE
020  TERMINAL LINESIZE(80)
030  WRITE ENTER NAME (AFTER [redacted]) OF FILE TO BE PUNCHED TODAY
040  WRITE
050  WRITENR ==>
060  READ &A
070  WRITE
080  COPY [redacted].PUNCH.CNTL SPARE.CNTL
090  EDIT SPARE CNTL OLD
100  C 10 170 /GOL/&A/ALL
110  UNNUM
120  SAVE
130  END
140  SUBMIT SPARE
150  DELETE SPARE
160  WRITE
170  WRITE FILE SENT TO BE PUNCHED TODAY
180  WRITE
190  STATUS [redacted]
200  WRITE
210  END
```

[redacted].[redacted].PUNCH.CNTL

```
//[redacted] JOB ([redacted],RJ013,,,,,1,,0),'[redacted]',
// CLASS=C,MSGLEVEL=0,TIME=(,20),PRTY=3,NOTIFY=[redacted]
/*ROUTE PRINT RMT13
/*ROUTE PUNCH LOCAL
//PUNCH EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT2 DD SYSOUT=B,DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//SYSUT1 DD DSN=[redacted].[redacted].GOL,DISP=SHR
/*
//PRINT EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT2 DD SYSOUT=A,DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//SYSUT1 DD DSN=[redacted].[redacted].GOL,DISP=SHR
/*
//
```

II-A.1.d. PRINT.CLIST - Print a TSO File

Similar to PUNCH.CLIST, this program prints out a TSO file. The user is prompted for the name.

[redacted].[redacted].PRINT.CLIST

```
010  WRITE
020  WRITE ENTER NAME (AFTER [redacted]) OF PROGRAM TO BE PRINTED
030  WRITE
040  WRITENR ===>
050  READ &X
060  WRITE
070  PRINT2 [redacted].[redacted].&X
080  WRITE
130  END
```

II-A.2. Job Control Programs for Operating a GOL Model

The GOL model is designed to operate in the TROLL simulation package. Although TROLL is an interactive system and can be operated accordingly, programs of TROLL commands can be submitted in the batch mode from cards or from TSO card image files. The card image files can be created by TSO editing routines, by punched cards, or by micro computer files telecommunicated to TSO. This section of the report lists some programs which allow inputs to GOL models in TROLL to be made in different ways.

II-A.2.a. Program to Submit a TROLL Card File for Batch Processing

This job control program submits TROLL commands as a batch job. The MACRO facility of TROLL is used to process the commands. This gives the advantage that unnecessary printout of 'interactive' TROLL commands is avoided and serious TROLL errors will terminate TROLL processing.

```
ID RSCS
WCC1 JOB
//[redacted] JOB ([redacted],RJ013),'GOL [redacted]',CLASS=C,
//  MSGLEVEL=1,TIME=(0,20),PRTY=3,NOTIFY=[redacted]
//*LOGONID [redacted]
//*PASSWORD [redacted]
//STEP1 EXEC ERSTROLL
//FT05F001 DD *
BATCH
CARDREAD;
OPRTMACRO GOLSUB ;
&GOLSUB
DELETE MACRO GOLSUB;
LOGOUT;
/*
//FT10F001 DD DSN=[redacted].[redacted].GOL.TROLL.DATA,
// UNIT=SYSDA,DISP=OLD
//FT04F001 DD *
TROLL_MACRO_GOLSUB
&ERROR &IGNORE
OUTOPT RMARG 88 FPTFIELD 14 TARWIDTH 4 ;
USEMOD RWWGOL;
MODEDIT;
PRINT ALL COMMENT;
QUIT;
/*
//
//
```

II-A.2.b. TROLL.CLIST - Program for Selecting and Submitting a TSO TROLL Program for Batch Processing from the TSO Interactive Mode

This program submits a selected file to TROLL in the batch mode. The CLIST calls appropriate CNTL files of job control commands. A listing of an interactive session illustrates the use of the program.

[redacted].[redacted].TROLL.CLIST

```
010  WRITE
020  WRITE ENTER NAME OF FILE TO GO TO TROLL
030  WRITE (E.G. USGOLEQC, ECGOLSMC, ETC.)
040  WRITE
050  WRITENR ===>
060  READ &A
070  WRITE
080  WRITE ENTER WIDTH OF CARD IMAGE FILE (72 OR 80)
090  WRITE
100  WRITENR ===>
110  READ &W
120  WRITE
130  WRITE ENTER NAME OF TROLL FILE TO BE USED
140  WRITE (E.G. GOL, USGOL, ETC.)
150  WRITE
160  WRITENR ===>
170  READ &B
180  WRITE
190  WRITE ENTER PRIORITY (2=NIGHT OR 3=DAY)
200  WRITE
210  WRITENR ===>
220  READ &P
230  WRITE
240  COPY [redacted].MACRO [redacted].BACK.&A NONUM
250  EDIT [redacted].BACK.&A DATA OLD
260  MERGE [redacted].&A NONUM
270  SAVE
280  END
290  COPY [redacted].TROLL&W&P..CNTL SPARE.CNTL
300  EDIT SPARE CNTL OLD
310  C 180 /MODEL/&B/
320  C 210 /SHR/(OLD,DELETE)/
330  C 80 /PROGRAM/&A/
340  C 200 /PROGRAM/BACK.&A/
350  UNNUM
360  SAVE
370  END
380  SUBMIT SPARE
390  DELETE SPARE
400  WRITE
410  WRITE FILE SUBMITTED TO BATCH
420  WRITE
430  STATUS [redacted]
440  WRITE
450  END
```

```
exec [redacted].troll

ENTER NAME OF FILE TO GO TO TROLL
(E.G. USGOLEQC, ECGOLSMC, ETC.)

--->temp

ENTER WIDTH OF CARD IMAGE FILE (72 OR 80)

--->72

ENTER NAME OF TROLL FILE TO BE USED
(E.G. GOL, USGOL, ETC.)

--->col

ENTER PRIORITY (2=NIGHT OR 3=DAY)

--->2

DATA SET [redacted].PACK.TEMP.DATA NOT LINE NUMBERED, USING NONUM
JOB [redacted](JOB02996) SUBMITTED  **  FREE ALL FILES  **
ENTRY (A) [redacted].SPARE.CNTL DELETED

FILE SUBMITTED TO BATCH

JOB [redacted](JOB02996) WAITING FOR EXECUTION, IN HOLD STATUS

READY
```

[redacted].[redacted].MACRO.DATA

```
TROLL_MACRO_GOLSUB
&ERROR &IGNORE
```

[redacted].[redacted].TROLL723.CNTL

```
//[redacted] JOB ([redacted],RJ013),'[redacted]',CLASS=C,
// MSGLEVEL=0,TIME=(0,20),PRTY=3,NOTIFY=[redacted]
/*ROUTE PRINT RMT13
//STEP1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT2 DD SYSOUT=A,DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//SYSUT1 DD DSN=[redacted],[redacted],PROGRAM,DATA,UNIT=SYSDA,DISP=SHR
/*
//STEP2 EXEC ERSTRL
//FT05F001 DD *
BATCH
CARDREAD NOSEQ ;
&GOLSUB
DELETE MACRO GOLSUB;
LOGOUT;
/*
//FT10F001 DD DSN=[redacted],[redacted],MODEL,TROLL,DATA,
// UNIT=SYSDA,DISP=SHR
//FT04F001 DD DSN=[redacted],[redacted],PROGRAM,DATA,
// UNIT=SYSDA,DISP=SHR
/*
//
```

[redacted].[redacted].TROLL722.CNTL

```
//[redacted] JOB ([redacted],RJ013),'[redacted]',CLASS=D,
// MSGLEVEL=0,TIME=(1,00),PRTY=2,NOTIFY=[redacted]
/*ROUTE PRINT RMT13
//STEP1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT2 DD SYSOUT=A,DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//SYSUT1 DD DSN=[redacted],[redacted],PROGRAM,DATA,UNIT=SYSDA,DISP=SHR
/*
//STEP2 EXEC ERSTRL
//FT05F001 DD *
BATCH
CARDREAD NOSEQ ;
&GOLSUB
DELETE MACRO GOLSUB;
LOGOUT;
/*
//FT10F001 DD DSN=[redacted],[redacted],MODEL,TROLL,DATA,
// UNIT=SYSDA,DISP=SHR
//FT04F001 DD DSN=[redacted],[redacted],PROGRAM,DATA,
// UNIT=SYSDA,DISP=SHR
/*
//
```

```
[redacted].[redacted].TROLL802.CNTL


//[redacted] JOB ([redacted],RJ013),'[redacted]',CLASS=D,
// MSGLEVEL=0,TIME=(1,00),PRTY=2,NOTIFY=[redacted]
/*ROUTE PRINT RMT13
//STEP1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT2 DD SYSOUT=A,DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//SYSUT1 DD DSN=[redacted].[redacted].PROGRAM.DATA,UNIT=SYSDA,DISP=SHR
/*
//STEP2 EXEC ERSTRL
//FT05F001 DD *
BATCH
CARDREAD;
&GOLSUB
DELETE MACRO GOLSUB;
LOGOUT;
/*
//FT10F001 DD DSN=[redacted].[redacted].MODEL.TROLL.DATA,
// UNIT=SYSDA,DISP=SHR
//FT04F001 DD DSN=[redacted].[redacted].PROGRAM.DATA,
// UNIT=SYSDA,DISP=SHR
/*
//


ID RSCS
MCC1 JOB
//[redacted] JOB ([redacted],RJ013),'GOL [redacted]',CLASS=C,
//  MSGLEVEL=1,TIME=(0,20),PRTY=3,NOTIFY=[redacted]
//*LOGONID  [redacted]
//*PASSWORD [redacted]
//STEP0 EXEC PGM=IKJEFT01,DYNAMNBR=50,REGION=500K
//SYSTSPRT DD SYSOUT=A
//SYSTSIN DD *
 DEL '[redacted].[redacted].TEMP.DATA'
/*
//STEP1 EXEC PGM=AR233,COND=(17,LT)
//MAGICMSG DD SYSOUT=A
//MAGICIN DD *
/*
//MAGICOUT DD UNIT=SYSDA,DISP=(NEW,CATLG),SPACE=(4000,(20,5),RLSE),
//  DCB=(RECFM=FB,LRECL=80,BLKSIZE=4000),
//  DSN=[redacted].[redacted].TEMP.DATA
/*
//PRINT EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT2 DD SYSOUT=A
//SYSUT1 DD DSN=[redacted].[redacted].TEMP.DATA,DISP=SHR
/*
//
```

II-A.3. Routine to Create a TROLL File - TROLLFRM

This interactive CLIST creates and allocates a TROLL file which can hold GOL component models.

```
READY
trollfrm
************************************************************

     YOU ARE USING THE  "TROLL"  FORM FILE CREATION
                     SYSTEM

                (OSTROLL)

      YOU CAN NOT USE  TROLL  UNLESS YOU PREALLOCATE A
      USER OWNED FILE IN THIS CLIST.


     IF ANY QUESTIONS SEE:    BOB OTTO   ROOM 196    7-2623
************************************************************

ENTER YOUR FULL TROLL DATA FILE NAME (NO QUOTES)
[redacted].[redacted].usgol.troll.data

ENTER THE NUMBER OF BLOCKS TO BE ALLOCATED
3000
                                DATA GENERATOR OUTPUT FOR CONTROL CARDS AND MESSAG
ES                          PAGE  0001
 dsd output=(trlfile)
          DSD OUTPUT=(TRLFILE)
 create quantity=3000
          CREATE QUANTITY=3000
/*
usgol
IEB700I                       DATA GENERATION HAS BEEN SUCCESSFULLY COMPLETED COMP
LETION CODE IS ZERO
FILE SYSOUT NOT FREED, IS NOT ALLOCATED
FILE FT05F001 NOT FREED, IS NOT ALLOCATED
FILE FT06F001 NOT FREED, IS NOT ALLOCATED
FILE FT10F001 NOT FREED, IS NOT ALLOCATED
FORMATTING AN OSTROLL FILESYSTEM
(ASSIGNED TO UNIT 10)
ENTER TROLL USERNAME (A8):
ENTER MAXIMUM RECORD COUNT (I5):
03000
FORMATTED  3000 RECORDS FOR USGOL

************************************************************
READY
```

II-B. Micro Computer Programs for Building Programs of TROLL Commands

The micro computer can be used as a cost effective tool to create GOL model equations, sets of TROLL commands, and data for TROLL GOL data files. Most of the programs of TROLL commands listed so far in this report can be created via micro computer programs that follow. The basic idea is to take advantage of standard GOL structures to combine 'template' TROLL commands with real model constants to create groups of TROLL commands and model equations. Some programs will be illustrated with input files and output files created by the programs. When the TROLL programs have been created (and/or modified by word processing programs on the micro) they can be telecommunicated to TSO files for subsequent execution.

The programs are written in Microsoft (trademark) BASIC and can be execute on any micro computer with a CP/M (trademark) operating system. Remarks in the programs indicate where specific tasks are carried out. Matrices of constants or numbers required by the programs can be created by spreadsheet programs such as SuperCalc (trademark). The micro computer file conventions follow those of the above named systems.

II-B.1. Programs to Create TROLL Commands

These programs actually create TROLL statements which perform specific tasks.

II-B.1.a. EQWRITE - Write out TROLL Equations from a Template and an Elasticity Matrix

This program reads a 'template' equation file and a matrix of elasticities associated with the file and writes out a) TROLL equations for the commodities listed in the matrix, b) CEDIT commands to change the elasticities to the values listed in the matrix, and c) the TROLL 'DO' equations to calculate intercepts for the equations. Once the syntax is in the 'template' equation and matrix is correct, the TROLL syntax in all of the created equations will be correct. The program has the option of not writing out terms for constants with zero values in the elasticity matrix. This can mean a much simpler model structure when equation systems can be represented by sparse matrices.

```
10 CLEAR 5000
20 REM CREATED BY VERNON OLEY RONINGEN - FEB. 12, 1983
30 REM FILE ----.TXT MUST HAVE:
40 REM LINE 1 FILE NAME (READ BUT NOT USED IN PROGRAM)
50 REM LINE 2 VARIABLE NAME - E.G. USQD$$
60 REM LINE 3-N (LIMIT N=10) EQUATION TERMS
70 REM       INCLUDING ELASTICITIES AND OPERATORS
80 REM       E.G. *(USPD$$'N/USPNG'X)**USQD$$##'C
90 REM       WHERE $$ IS OWN COMMODITY CODE AND ## IS
100 REM       CROSS COMMODITY CODE.
110 REM
120 REM FILE ----.PRN MUST HAVE MATRIX OF ELASTICITIES;
130 REM       OWN, CROSS, AND OTHER TERMS IN ORDER OF THE
140 REM       TERMS IN THE VARIABLE FILE
150 REM       NOTE THAT COMMODITY CODES MUST BE RIGHT JUSTIFIED IN
160 REM       FIELDS IN THE COLUMN HEADS AND ROW LABELS
170 REM
180 REM ----------SAMPLE INPUT FILES----------------------------------------
190 REM --------------------------------------------------------------------
200 REM LINE 1 - MD.TXT
210 REM LINE 2 - MD$$
220 REM LINE 3 - *(PNG'X/PD$$'N)**MD$$##PC'C
230 REM LINE 4 - *(PNG'X(-1)/PD$$'N(-1))**MD$$##PL'C
240 REM LINE 5 - PD$$'N
250 REM --------------------------------------------------------------------
260 REM LINE 1 - MD.PRN            BF        PK
270 REM LINE 2 -
280 REM LINE 3 -          BF       .37       0.0
290 REM LINE 4 -          PK       .27       0.0
300 REM --------------------------------------------------------------------
```

```
310 GOSUB 2280
320 PRINT " E Q W R I T E   BY VOR":PRINT
330 PRINT"PROGRAM TO WRITE A SIMPLIFIED GOL EQUATION SET":PRINT
340 W$="":R9=0:INPUT"DO YOU WANT TO THIN THE EQUATIONS (Y OR N)";W$
350 PRINT:IF W$="N" THEN R9=1
360 R7=1:PRINT:W$="":INPUT"DO YOU WANT TO WRITE THE COEFFICIENT CREATION CARDS (Y OR N)";W
370 PRINT:IF W$="N"THEN R7=0
380 INPUT"ENTER NAME OF EQUATION FILE (.TXT FILE) ON DRIVE 'B'";W$:PRINT
390 REM ----------READ IN EQUATION (**.TXT) FILE------------------------------
400 PRINT"READING EQUATION FILE":PRINT
410 F$="B:"+W$+".TXT":OPEN"I",1,F$
420 IF R7=1 THEN GOTO 430 ELSE GOTO 440
430 F$="B:"+W$+"C.TXT":OPEN"O",2,F$
440 DIM V$(1,10):L0=0
450 LINE INPUT#1,X$
460 L0=L0+1
470 LINE INPUT#1,V$(1,L0)
480 IF EOF(1)THEN GOTO 500
490 PRINT V$(1,L0):GOTO 460
500 PRINT V$(1,L0):CLOSE 1
510 I8=0
520 REM ----------READ IN ELASTICITY (**.PRN) FILE----------------------------
530 PRINT:F$="B:"+W$+".PRN":PRINT"READING ELASTICITY FILE":PRINT
540 OPEN"I",1,F$:F$="B:"+W$+"R.TXT"
550 DIM C$(1,25),D(19,25),R$(1,19)
560 LINE INPUT#1,W$:J0=LEN(W$)/9-1
570 FOR J=1 TO J0:U$=MID$(W$,J*9+1,9):GOSUB 920
580 C$(1,J)=U$
590 PRINT"COLUMN LABEL ";J,C$(1,J)
600 NEXT J:I0=0
610 LINE INPUT#1,W$
620 PRINT
630 LINE INPUT#1,W$:I0=I0+1
640 IF EOF(1) THEN CLOSE 1:I8=1
650 U$=LEFT$(W$,9):GOSUB 920
660 R$(1,I0)=U$
670 PRINT"ROW (COMMODITY) LABEL ";I0,R$(1,I0)
680 FOR J=1 TO J0
690 U$=MID$(W$,J*9+1,9)
700 GOSUB 920
710 D(I0,J)=VAL(U$)
720 NEXT J
730 IF I0=19 AND J0=13 THEN GOTO 740 ELSE GOTO 890
740 LINE INPUT#1,W$:IF EOF(1) THEN CLOSE 1:I8=1:GOTO 890
750 I8=0
760 J0=LEN(W$)/9:PRINT
770 FOR J=1 TO J0:U$=MID$(W$,(J-1)*9+1,9):GOSUB 920
780 C$(1,J+13)=U$
790 PRINT"COLUMN LABEL ";J+13,C$(1,J+13)
800 NEXT J:I0=0:LINE INPUT#1,W$
810 LINE INPUT #1,W$:I0=I0+1
820 IF EOF(1) THEN CLOSE 1:I8=1
830 FOR J = 1 TO J0
840 U$=MID$(W$,(J-1)*9+1,9):GOSUB 920
850 D(I0,J+13)=VAL(U$)
860 NEXT J
870 IF I8=1 THEN J0=13+J0
880 IF I8=0 THEN GOTO 810
890 IF I8=0 THEN GOTO 630 ELSE GOTO 1030
900 GOTO 1030
910 REM ----------SUBROUTINE TO REMOVE LEFT BLANKS FROM U$
920 Z$=U$
930 IF LEFT$(Z$,1)=" "THEN Z$=RIGHT$(Z$,LEN(Z$)-1) ELSE 950
940 GOTO 930
950 U$=Z$:RETURN
960 REM ----------SUBROUTINE TO REMOVE LEFT BLANKS FROM W$
970 IF LEFT$(W$,1)=" "THEN GOTO 980 ELSE GOTO 990
980 W$=RIGHT$(W$,LEN(W$)-1):GOTO 970
990 RETURN
```

```
1000 PRINT"---------+---------+---------+---------+---------+---------+"
1010 PRINT#1,"---------+---------+---------+---------+---------+---------+---------"
1020 RETURN
1030 GOSUB 2280
1040 REM ----------WRITE MODEL EQUATIONS----------------------------------------
1050 OPEN"O",1,F$:PRINT"WRITE EQUATIONS":PRINT:J5=0
1060 FOR I=1 TO I0:IF R$(1,I)=C$(1,I) THEN J5=J5+1:NEXT I
1070 PRINT"BEGINNING OF FILE":PRINT#1,"BEGINNING OF FILE"
1080 GOSUB 1000
1090 FOR I=1 TO I0:L$=""
1100 W$=V$(1,1)+": "+V$(1,1)+"'N = "+V$(1,1)+"I'C"
1110 X$="$$":Y$=R$(1,I):GOSUB 1990
1120 F=1:GOSUB 2030
1130 IF R9=0 AND D(I,1)=0 THEN GOTO 1170
1140 W$=V$(1,2):X$="$$":Y$=R$(1,I):GOSUB 1990
1150 X$="##":Y$=C$(1,1):GOSUB 1990
1160 F=1:GOSUB 2030
1170 IF J5<2 THEN GOTO 1240
1180 FOR J=2 TO J5
1190 IF R9=0 AND D(I,J)=0 THEN GOTO 1230
1200 W$=V$(1,2):X$="$$":Y$=R$(1,I):GOSUB 1990
1210 X$="##":Y$=C$(1,J):GOSUB 1990
1220 F=1:GOSUB 2030
1230 NEXT J
1240 FOR L=3 TO L0
1250 IF J5=0 THEN J=L+J5-1 ELSE J=L+J5-2
1260 W$=V$(1,L):X$="$$":Y$=R$(1,I):GOSUB 1990
1270 IF J>J0 THEN GOTO 1300
1280 IF R9=0 AND D(I,J)=0 THEN GOTO 1310
1290 X$="##":Y$=C$(1,J):GOSUB 1990
1300 F=1:GOSUB 2030
1310 NEXT L
1320 W$=",":F=1:GOSUB 2030
1330 PRINT L$:PRINT#1,L$:NEXT I
1340 GOSUB 2280
1350 REM ----------WRITE COEFFICIENTS-----------------------------------------
1360 PRINT"WRITE COEFFICIENTS":PRINT
1370 GOSUB 1000
1380 IF R7=1 THEN PRINT#2,"BEGINNING OF FILE"
1390 IF R7=1 THEN L$=""
1400 X$=LEFT$(V$(1,1),4)
1410 IF R7=1 THEN GOTO 1420 ELSE GOTO 1450
1420 FOR I=1 TO I0
1430 W$=X$+R$(1,I)+"I'C 0, ":F=2:GOSUB 2030
1440 NEXT I
1450 FOR J=1 TO J0
1460 FOR I=1 TO I0
1470 IF R9=0 AND D(I,J)=0 THEN GOTO 1520
1480 W$=X$+R$(1,I)+C$(1,J)+"  "+STR$(D(I,J))+","
1490 PRINT W$:PRINT#1,W$
1500 IF R7=1 THEN GOTO 1510 ELSE GOTO 1520
1510 W$=X$+R$(1,I)+C$(1,J)+" 0, ":F=2:GOSUB 2030
1520 NEXT I,J
1530 IF R7=1 THEN GOTO 1540 ELSE GOTO 1570
1540 PRINT#2,L$:PRINT#2,"END OF FILE":CLOSE 2
1550 GOSUB 2280
1560 REM ----------WRITE INTERCEPT CALCULATION STATEMENTS---------------------
1570 PRINT"WRITE INTERCEPT CALCULATION STATEMENTS":PRINT
1580 GOSUB 1000
1590 FOR I=1 TO I0:L$=""
1600 W$="DOCORE INTERCPT="+V$(1,1)+"/(":X$="$$":Y$=R$(1,I):GOSUB 1990
1610 GOSUB 2140
1620 F=1:GOSUB 2030
1630 K1=0
1640 IF R9=0 AND D(I,1)=0 THEN GOTO 1700
1650 W$=V$(1,2):X$="$$":Y$=R$(1,I):GOSUB 1990
1660 W$=RIGHT$(W$,LEN(W$)-1):K1=1
1670 X$="##":Y$=C$(1,1):GOSUB 1990
1680 GOSUB 2140
1690 F=1:GOSUB 2030
1700 IF J5<2 THEN GOTO 1800
1710 FOR J=2 TO J5
1720 IF R9=0 AND D(I,J)=0 THEN GOTO 1790
```

```
1730 W$=V$(1,2):X$="$$":Y$=R$(1,I):GOSUB 1990
1740 IF K1=0 THEN GOTO 1750 ELSE GOTO 1760
1750 W$=RIGHT$(W$,LEN(W$)-1):K1=1
1760 X$="##":Y$=C$(1,J):GOSUB 1990
1770 GOSUB 2140
1780 F=1:GOSUB 2030
1790 NEXT J
1800 FOR L=3 TO L0
1810 IF J5=0 THEN J=L+J5-1 ELSE J=L+J5-2
1820 W$=V$(1,L):X$="$$":Y$=R$(1,I):GOSUB 1990
1830 IF J>J0 THEN GOTO 1860
1840 IF R9=0 AND D(I,J)=0 THEN GOTO 1900
1850 X$="##":Y$=C$(1,J):GOSUB 1990
1860 IF K1=0 THEN GOTO 1870 ELSE GOTO 1880
1870 W$=RIGHT$(W$,LEN(W$)-1):K1=1
1880 GOSUB 2140
1890 F=1:GOSUB 2030
1900 NEXT L
1910 W$=");":F=1:GOSUB 2030
1920 PRINT L$:PRINT#1,L$:L$=""
1930 W$="DO "+V$(1,1)+"I'C=MEAN(INTERCPT);":X$="$$":Y$=R$(1,I):GOSUB 1990
1940 F=1:GOSUB 2030
1950 PRINT L$:PRINT#1,L$:NEXT I
1960 GOSUB 1000
1970 PRINT"END OF FILE":PRINT#1,"END OF FILE":CLOSE 1:END
1980 REM ----------SUBROUTINE TO REPLACE STRING X$ WITH STRING Y$ IN STRING W$-
1990 P0=INSTR(W$,X$)
2000 IF P0=0 THEN RETURN
2010 MID$(W$,P0)=Y$:GOTO 1990
2020 REM ----------SUBROUTINE TO PRINT LINE IF NEW ONE > 72 CHARS.-------------
2030 IF LEN(L$)+LEN(W$)<=72 THEN GOTO 2050 'PRINT IF LEN>80
2040 PRINT L$:PRINT#F,L$:L$=""
2050 L$=L$+W$:RETURN
2060 REM ----------SUBROUTINE TO REMOVE STRING X$ FROM STRING W$---------------
2070 P0=INSTR(W$,X$)
2080 IF P0=0 THEN RETURN
2090 IF P1=1 THEN W$=RIGHT$(W$,LEN(W$)-LEN(X$))
2100 IF P0=LEN(W$)-LEN(X$)+1 THEN W$=LEFT$(W$,LEN(W$)-LEN(X$)) ELSE GOTO 2130
2110 GOTO 2070
2120 REM ----------SUBROUTINE TO REMOVE SUFFIXES FROM INTERCEPT 'DO' STATEMENTS
2130 W$=LEFT$(W$,P0-1)+RIGHT$(W$,LEN(W$)-LEN(X$)-P0+1):GOTO 2070
2140 X$="'POLN":GOSUB 2070
2150 X$="'POLP":GOSUB 2070
2160 X$="'NPOL":GOSUB 2070
2170 X$="'DEF":GOSUB 2070
2180 X$="'XNS":GOSUB 2070
2190 X$="'NXS":GOSUB 2070
2200 X$="'POL":GOSUB 2070
2210 X$="'XN":GOSUB 2070
2220 X$="'XP":GOSUB 2070
2230 X$="'NX":GOSUB 2070
2240 X$="'N":GOSUB 2070
2250 X$="'X":GOSUB 2070
2260 RETURN
2270 REM ----------SUBROUTINE TO CLEAR SCREEN----------------------------------
2280 PRINT CHR$(27);CHR$(58):RETURN
```

The purpose of the program EQWRITE is, given an appropriate input 'template' equation and matrix of elasticities, to write out the full set of TROLL equations for all GOL commodities, the full set of elasticity entry statements and the full set of TROLL 'DO' statements needed to calculate the intercepts for these equations. This amounts to writing out most of the TROLL commands concerned with any set of equations from a minimum amount of information about the equation structure in the template equation and information about the elasticities in the input matrix.

Input equation template file.

```
USMD.TXT
USMD$$
*(USPNG'X/USPD$$'N)**USMD$$##'C
*(USPNG'X(-1)/USPD$$'N(-))**USMD$$##'C
*USPD$$'N
```

Input matrix file of elasticities.

USMD.PRN	PC	PL
BF	.37	0
PK	.27	0
ML	0	0
DM	0	0
PM	0	0
PE	0	0
WH	0	0
CN	1	0
CG	1	0
RI	0	0
SB	0	0
OS	0	0
SM	1	0
SO	.02	0
OM	1	.4
OO	0	0
DB	0	0
DC	0	0
DO	0	0

The $$'s in the equation template are replaced by commodity codes in the left row of the input matrix while the ##'s are replaced by the codes over the columns of the matrix. The program gives the user an option of excluding equation terms with zero valued coefficients.

Example output file 1.

TROLL equations

```
BEGINNING OF FILE
---------+---------+---------+---------+---------+---------+---------+---------+
USMDBF: USMDBF'N = USMDBFI'C*(USPNG'X/USPDBF'N)**USMDBFPC'C
*(USPNG'X(-1)/USPDBF'N(-))**USMDBFPL'C,
USMDPK: USMDPK'N = USMDPKI'C*(USPNG'X/USPDPK'N)**USMDPKPC'C
*(USPNG'X(-1)/USPDPK'N(-))**USMDPKPL'C,
USMDML: USMDML'N = USMDMLI'C*(USPNG'X/USPDML'N)**USMDMLPC'C
*(USPNG'X(-1)/USPDML'N(-))**USMDMLPL'C,
USMDDM: USMDDM'N = USMDDMI'C*(USPNG'X/USPDDM'N)**USMDDMPC'C
*(USPNG'X(-1)/USPDDM'N(-))**USMDDMPL'C,
USMDPM: USMDPM'N = USMDPMI'C*(USPNG'X/USPDPM'N)**USMDPMPC'C
*(USPNG'X(-1)/USPDPM'N(-))**USMDPMPL'C,
USMDPE: USMDPE'N = USMDPEI'C*(USPNG'X/USPDPE'N)**USMDPEPC'C
*(USPNG'X(-1)/USPDPE'N(-))**USMDPEPL'C,
USMDWH: USMDWH'N = USMDWHI'C*(USPNG'X/USPDWH'N)**USMDWHPC'C
*(USPNG'X(-1)/USPDWH'N(-))**USMDWHPL'C,
USMDCN: USMDCN'N = USMDCNI'C*(USPNG'X/USPDCN'N)**USMDCNPC'C
*(USPNG'X(-1)/USPDCN'N(-))**USMDCNPL'C,
USMDCG: USMDCG'N = USMDCGI'C*(USPNG'X/USPDCG'N)**USMDCGPC'C
*(USPNG'X(-1)/USPDCG'N(-))**USMDCGPL'C,
USMDRI: USMDRI'N = USMDRII'C*(USPNG'X/USPDRI'N)**USMDRIPC'C
*(USPNG'X(-1)/USPDRI'N(-))**USMDRIPL'C,
USMDSB: USMDSB'N = USMDSBI'C*(USPNG'X/USPDSB'N)**USMDSBPC'C
*(USPNG'X(-1)/USPDSB'N(-))**USMDSBPL'C,
USMDOS: USMDOS'N = USMDOSI'C*(USPNG'X/USPDOS'N)**USMDOSPC'C
*(USPNG'X(-1)/USPDOS'N(-))**USMDOSPL'C,
USMDSM: USMDSM'N = USMDSMI'C*(USPNG'X/USPDSM'N)**USMDSMPC'C
*(USPNG'X(-1)/USPDSM'N(-))**USMDSMPL'C,
USMDSO: USMDSO'N = USMDSOI'C*(USPNG'X/USPDSO'N)**USMDSOPC'C
*(USPNG'X(-1)/USPDSO'N(-))**USMDSOPL'C,
USMDOM: USMDOM'N = USMDOMI'C*(USPNG'X/USPDOM'N)**USMDOMPC'C
*(USPNG'X(-1)/USPDOM'N(-))**USMDOMPL'C,
USMDOO: USMDOO'N = USMDOOI'C*(USPNG'X/USPDOO'N)**USMDOOPC'C
*(USPNG'X(-1)/USPDOO'N(-))**USMDOOPL'C,
USMDDB: USMDDB'N = USMDDBI'C*(USPNG'X/USPDDB'N)**USMDDBPC'C
*(USPNG'X(-1)/USPDDB'N(-))**USMDDBPL'C,
USMDDO: USMDDO'N = USMDDOI'C*(USPNG'X/USPDDO'N)**USMDDOPC'C
*(USPNG'X(-1)/USPDDO'N(-))**USMDDOPL'C,
---------+---------+---------+---------+---------+---------+---------+---------+
```

```
USMDBFPC    .37,
USMDPKPC    .27,
USMDMLPC    0,
USMDDMPC    0,
USMDPMPC    0,
USMDPEPC    0,
USMDWHPC    0,
USMDCNPC    1,
USMDCGPC    1,
USMDRIPC    0,
USMDSBPC    0,
USMDOSPC    0,
USMDSMPC    1,
USMDSOPC    .02,
USMDOMPC    1,
USMDOOPC    0,
USMDDBPC    0,
USMDDOPC    0,
USMDBFPL    0,
USMDPKPL    0,
USMDMLPL    0,
USMDDMPL    0,
USMDPMPL    0,
USMDPEPL    0,
USMDWHPL    0,
USMDCNPL    0,
USMDCGPL    0,
USMDRIPL    0,
USMDSBPL    0,
USMDOSPL    0,
USMDSMPL    0,
USMDSOPL    0,
USMDOMPL    .4,
USMDOOPL    0,
USMDDBPL    0,
USMDDOPL    0,
```

← Elasticity entry commands

Equations to calculate intercepts from elasticities and historical data

↓

```
---------+---------+---------+---------+---------+---------+---------+---------+
DOCORE INTERCPT=USMDBF/((USPNG/USPDBF)**USMDBFPC'C
*(USPNG(-1)/USPDBF(-))**USMDBFPL'C);
DO USMDBFI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDPK/((USPNG/USPDPK)**USMDPKPC'C
*(USPNG(-1)/USPDPK(-))**USMDPKPL'C);
DO USMDPKI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDML/((USPNG/USPDML)**USMDMLPC'C
*(USPNG(-1)/USPDML(-))**USMDMLPL'C);
DO USMDMLI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDDM/((USPNG/USPDDM)**USMDDMPC'C
*(USPNG(-1)/USPDDM(-))**USMDDMPL'C);
DO USMDDMI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDPM/((USPNG/USPDPM)**USMDPMPC'C
*(USPNG(-1)/USPDPM(-))**USMDPMPL'C);
DO USMDPMI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDPE/((USPNG/USPDPE)**USMDPEPC'C
*(USPNG(-1)/USPDPE(-))**USMDPEPL'C);
DO USMDPEI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDWH/((USPNG/USPDWH)**USMDWHPC'C
*(USPNG(-1)/USPDWH(-))**USMDWHPL'C);
DO USMDWHI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDCN/((USPNG/USPDCN)**USMDCNPC'C
*(USPNG(-1)/USPDCN(-))**USMDCNPL'C);
DO USMDCNI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDCG/((USPNG/USPDCG)**USMDCGPC'C
*(USPNG(-1)/USPDCG(-))**USMDCGPL'C);
DO USMDCGI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDRI/((USPNG/USPDRI)**USMDRIPC'C
*(USPNG(-1)/USPDRI(-))**USMDRIPL'C);
DO USMDRII'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDSB/((USPNG/USPDSB)**USMDSBPC'C
*(USPNG(-1)/USPDSB(-))**USMDSBPL'C);
DO USMDSBI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDOS/((USPNG/USPDOS)**USMDOSPC'C
*(USPNG(-1)/USPDOS(-))**USMDOSPL'C);
DO USMDOSI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDSM/((USPNG/USPDSM)**USMDSMPC'C
*(USPNG(-1)/USPDSM(-))**USMDSMPL'C);
DO USMDSMI'C=MEAN(INTERCPT);
```

```
DOCORE INTERCPT=USMDSO/((USPNG/USPDSO)**USMDSOPC'C
*(USPNG(-1)/USPDSO(-))**USMDSOPL'C);
DO USMDSOI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDOM/((USPNG/USPDOM)**USMDOMPC'C
*(USPNG(-1)/USPDOM(-))**USMDOMPL'C);
DO USMDOMI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDOO/((USPNG/USPDOO)**USMDOOPC'C
*(USPNG(-1)/USPDOO(-))**USMDOOPL'C);
DO USMDOOI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDDB/((USPNG/USPDDB)**USMDDBPC'C
*(USPNG(-1)/USPDDB(-))**USMDDBPL'C);
DO USMDDBI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDDO/((USPNG/USPDDO)**USMDDOPC'C
*(USPNG(-1)/USPDDO(-))**USMDDOPL'C);
DO USMDDOI'C=MEAN(INTERCPT);
---------+---------+---------+---------+---------+---------+---------+---------+
END OF FILE
```

Elasticity creation commands

```
BEGINNING OF FILE
USMDBFI'C 0, USMDPKI'C 0, USMDMLI'C 0, USMDDMI'C 0, USMDPMI'C 0,
USMDPEI'C 0, USMDWHI'C 0, USMDCNI'C 0, USMDCGI'C 0, USMDRII'C 0,
USMDSBI'C 0, USMDOSI'C 0, USMDSMI'C 0, USMDSOI'C 0, USMDOMI'C 0,
USMDOOI'C 0, USMDDBI'C 0, USMDDOI'C 0, USMDBFPC 0, USMDPKPC 0,
USMDMLPC 0, USMDDMPC 0, USMDPMPC 0, USMDPEPC 0, USMDWHPC 0, USMDCNPC 0,
USMDCGPC 0, USMDRIPC 0, USMDSBPC 0, USMDOSPC 0, USMDSMPC 0, USMDSOPC 0,
USMDOMPC 0, USMDOOPC 0, USMDDBPC 0, USMDDOPC 0, USMDBFPL 0, USMDPKPL 0,
USMDMLPL 0, USMDDMPL 0, USMDPMPL 0, USMDPEPL 0, USMDWHPL 0, USMDCNPL 0,
USMDCGPL 0, USMDRIPL 0, USMDSBPL 0, USMDOSPL 0, USMDSMPL 0, USMDSOPL 0,
USMDOMPL 0, USMDOOPL 0, USMDDBPL 0, USMDDOPL 0,
END OF FILE
```

Two examples of output files created from the two input files are presented.

The first example did not use the option of 'thinning' out the zero value coefficients while the second example exercised this option. Both give the coefficients for the elasticity creation programs. The outputs from this example are the basic ingredients for the domestic margin part of the file USGOLMA presented earlier in this report.

Example output file 2.

```
BEGINNING OF FILE
---------+---------+---------+---------+---------+---------+---------+---------+
USMDBF: USMDBF'N = USMDBFI'C*(USPNG'X/USPDBF'N)**USMDBFPC'C,
USMDPK: USMDPK'N = USMDPKI'C*(USPNG'X/USPDPK'N)**USMDPKPC'C,
USMDML: USMDML'N = USMDMLI'C,
USMDDM: USMDDM'N = USMDDMI'C,
USMDPM: USMDPM'N = USMDPMI'C,
USMDPE: USMDPE'N = USMDPEI'C,
USMDWH: USMDWH'N = USMDWHI'C,
USMDCN: USMDCN'N = USMDCNI'C*(USPNG'X/USPDCN'N)**USMDCNPC'C,
USMDCG: USMDCG'N = USMDCGI'C*(USPNG'X/USPDCG'N)**USMDCGPC'C,
USMDRI: USMDRI'N = USMDRII'C,
USMDSB: USMDSB'N = USMDSBI'C,
USMDOS: USMDOS'N = USMDOSI'C,
USMDSM: USMDSM'N = USMDSMI'C*(USPNG'X/USPDSM'N)**USMDSMPC'C,
USMDSO: USMDSO'N = USMDSOI'C*(USPNG'X/USPDSO'N)**USMDSOPC'C,
USMDOM: USMDOM'N = USMDOMI'C*(USPNG'X/USPDOM'N)**USMDOMPC'C
*(USPNG'X(-1)/USPDOM'N(-))**USMDOMPL'C,
USMDOO: USMDOO'N = USMDOOI'C,
USMDDB: USMDDB'N = USMDDBI'C,
USMDDO: USMDDO'N = USMDDOI'C,
```

```
---------+---------+---------+---------+---------+---------+---------+---------+
USMDBFPC    .37,
USMDPKPC    .27,
USMDCNPC    1,
USMDCGPC    1,
USMDSMPC    1,
USMDSOPC    .02,
USMDOMPC    1,
USMDOMPL    .4,
---------+---------+---------+---------+---------+---------+---------+---------+
DOCORE INTERCPT=USMDBF/((USPNG/USPDBF)**USMDBFPC'C);
DO USMDBFI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDPK/((USPNG/USPDPK)**USMDPKPC'C);
DO USMDPKI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDML/(0);
DO USMDMLI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDDM/(0);
DO USMDDMI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDPM/(0);
DO USMDPMI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDPE/(0);
DO USMDPEI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDWH/(0);
DO USMDWHI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDCN/((USPNG/USPDCN)**USMDCNPC'C);
DO USMDCNI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDCG/((USPNG/USPDCG)**USMDCGPC'C);
DO USMDCGI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDRI/(0);
DO USMDRII'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDSB/(0);
DO USMDSBI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDOS/(0);
DO USMDOSI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDSM/((USPNG/USPDSM)**USMDSMPC'C);
DO USMDSMI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDSO/((USPNG/USPDSO)**USMDSOPC'C);
DO USMDSOI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDOM/((USPNG/USPDOM)**USMDOMPC'C
*(USPNG(-1)/USPDOM(-))**USMDOMPL'C);
DO USMDOMI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDOO/(0);
DO USMDOOI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDDB/(0);
DO USMDDBI'C=MEAN(INTERCPT);
DOCORE INTERCPT=USMDDO/(0);
DO USMDDOI'C=MEAN(INTERCPT);
---------+---------+---------+---------+---------+---------+---------+---------+
END OF FILE



BEGINNING OF FILE
USMDBFI'C 0, USMDPKI'C 0, USMDMLI'C 0, USMDDMI'C 0, USMDPMI'C 0,
USMDPEI'C 0, USMDWHI'C 0, USMDCNI'C 0, USMDCGI'C 0, USMDRII'C 0,
USMDSBI'C 0, USMDOSI'C 0, USMDSMI'C 0, USMDSOI'C 0, USMDOMI'C 0,
USMDOOI'C 0, USMDDBI'C 0, USMDDOI'C 0, USMDBFPC 0, USMDPKPC 0,
USMDCNPC 0, USMDCGPC 0, USMDSMPC 0, USMDSOPC 0, USMDOMPC 0, USMDOMPL 0,
END OF FILE
```

II-B.1.b. EQDUPLIC - Write out TROLL Identity Equations from a Template and Equation Summary Matrix

TROLL equations which do not have constants or elasticities but do have different terms can be automatically created with this program. If a particular commodity does not require a particular term in its equation (a zero in the matrix) the term need not be written out in the equation. Between this program and EQWRITE, all GOL model equations can be automatically written out on a micro computer disk file once a template and input matrix have been created.

```
10 REM CREATED BY VERNON OLEY RONINGEN - FEB. 14, 1983
20 CLEAR 5000
30 REM THIS PROGRAM REQUIRES 2 INPUT FILES:
40 REM AN EQUATION MASTER FILE (--.TXT) AND A ZERO-ONE MATRIX FILE (--.PRN)
50 REM
60 REM FILE ----.TXT MUST HAVE:
70 REM LINE 1 FILE NAME
80 REM LINE 2 VARIABLE NAME - E.G. USQD$$
90 REM LINE 3 REST OF VARIABLE NAME AND EQUAL SIGN
100 REM      E.G. DEF == ABSV'F
110 REM LINE 4-N (LIMIT N=15) ONE VARIABLE IN EACH LINE
120 REM      E.G. -USQD$$'N
130 REM     REMEMBER TO INCLUDE ALL NECESSARY PARENTHESES WITH
140 REM     WITH VARIABLES OR ON EXTRA LINES
150 REM
160 REM FILE ----.PRN MUST HAVE:
170 REM LINE 1 FILE NAME PLUS DESIRED COLUMN LABELS
180 REM LINE 2 BLANK LINE
190 REM LINE 3-N ROWS FOR EACH GOL COMMODITY WITH A 1 OR 0 FOR EACH TERM OR
200 REM      EACH VARIABLE IN LINES 4-N, A '1' MEANS THAT
210 REM      THE VARIABLE SHOULD BE INCLUDED IN THE EQUATION
220 REM      A '0' MEANS THAT A VARIABLE IS NOT INCLUDED
230 REM
240 REM ----------SAMPLE INPUT FILES FOLLOW:
250 REM ------------------------------------------------------------------------
260 REM LINE 1 - PS.TXT
270 REM LINE 2 - PS$$
280 REM LINE 3 - 'DEF ==
290 REM LINE 4 - PD$$'N
300 REM LINE 5 - -MD$$'N
310 REM ------------------------------------------------------------------------
320 REM LINE 1 - PS.PRN        PS$$      MD$$
330 REM LINE 2 -
340 REM LINE 3 -        BF         1         1
350 REM LINE 4 -        PK         1         0
360 REM ------------------------------------------------------------------------
```

```
370 GOSUB 1590
380 PRINT" E Q D U P L I C   BY VOR":PRINT
390 PRINT"PROGRAM TO WRITE A SIMPLIFIED GOL EQUATION SET":PRINT
400 W$="":R9=1:INPUT"DO YOU WANT TO THIN THE EQUATIONS (Y OR N)";W$
410 PRINT:IF W$="N" THEN R9=0
420 INPUT"ENTER NAME OF EQUATION FILE (.TXT FILE)";W$:PRINT
430 REM ----------READ EQUATION MASTER FILE-----------------------------------
440 PRINT"READING EQUATION FILE":PRINT
450 F$="B:"+W$+".TXT":OPEN"I",1,F$
460 DIM V$(1,20):LO=0
470 I8=0
480 LINE INPUT#1,X$
490 LO=LO+1
500 LINE INPUT#1,V$(1,LO)
510 IF EOF(1)THEN GOTO 530
520 PRINT V$(1,LO):GOTO 490
530 PRINT V$(1,LO):CLOSE 1
540 REM ----------READ ZERO-ONE MATRIX FILE-----------------------------------
550 PRINT:F$="B:"+W$+".PRN":PRINT"READING MATRIX FILE":PRINT
560 OPEN"I",1,F$:F$="B:"+W$+"R.TXT"
570 DIM C$(1,25),D(19,25),R$(1,19)
580 LINE INPUT#1,W$:JO=LEN(W$)/9-1
590 FOR J=1 TO JO:U$=MID$(W$,J*9+1,9):GOSUB 770
600 C$(1,J)=U$
610 PRINT"COLUMN LABEL ";J,C$(1,J)
620 NEXT J:IO=0
630 LINE INPUT#1,W$
640 PRINT
650 LINE INPUT#1,W$:IO=IO+1
660 IF EOF(1) THEN CLOSE 1:I8=1
670 U$=LEFT$(W$,9):GOSUB 770
680 R$(1,IO)=U$
690 PRINT"ROW (COMMODITY) LABEL ";IO,R$(1,IO)
700 FOR J=1 TO JO
710 U$=MID$(W$,J*9+1,9)
720 GOSUB 770
730 D(IO,J)=VAL(U$)
740 NEXT J
750 IF I8 = 0 THEN GOTO 650 ELSE GOTO 930
760 REM ----------SUBROUTINE TO REMOVE LEADING AND TRAILING BLANKS FROM Z$----------
770 Z$=U$
780 IF LEFT$(Z$,1)=" "THEN Z$=RIGHT$(Z$,LEN(Z$)-1) ELSE 800
790 GOTO 780
800 IF RIGHT$(Z$,1)=" "THEN Z$=LEFT$(Z$,LEN(Z$)-1) ELSE 820
810 GOTO 800
820 U$=Z$:RETURN
830 REM ----------SUBROUTINE TO REMOVE LEADING BLANKS FROM W$------------------
840 IF LEFT$(W$,1)=" "THEN GOTO 850 ELSE GOTO 860
850 W$=RIGHT$(W$,LEN(W$)-1):GOTO 840
860 RETURN
870 REM ----------SUBROUTINE TO PRINT LINES ----------------------------------
880 PRINT"---------+---------+---------+---------+---------+---------+"
890 PRINT#1,"---------+---------+---------+---------+---------+---------+---------+-----
900 RETURN
910 GOSUB 1590
920 REM ----------WRITE TROLL EQUATIONS---------------------------------------
930 GOSUB 1590: PRINT:OPEN"O",1,F$:PRINT"WRITE EQUATIONS":PRINT:J5=0
940 FOR I=1 TO IO:IF R$(1,I)=C$(1,I) THEN J5=J5+1:NEXT I
950 PRINT"BEGINNING OF FILE":PRINT#1,"BEGINNING OF FILE"
960 GOSUB 880
970 FOR I=1 TO IO:L$=""
980 W$=V$(1,1)+": "+V$(1,1)+V$(1,2)
990 X$="$$":Y$=R$(1,I):GOSUB 1300
1000 GOSUB 1350
1010 FOR L=3 TO LO
1020 IF R9=1 AND D(I,L-2)=0 THEN GOTO 1050
1030 W$=V$(1,L):X$="$$":Y$=R$(1,I):GOSUB 1300
1040 GOSUB 1350
1050 NEXT L
1060 W$=",":GOSUB 1350
1070 PRINT L$:PRINT#1,L$:NEXT I
1080 GOSUB 1590
1090 W$="":PRINT:INPUT"DO YOU WANT DO EQUATIONS AS WELL (Y OR N)";W$
1100 IF W$="N" THEN GOTO 1280
```

```
1110 REM ----------WRITE TROLL 'DO' EQUATIONS----------------------------------
1120 PRINT:PRINT"WRITE 'DO' EQUATIONS":PRINT
1130 GOSUB 880
1140 REM
1150 FOR I=1 TO IO:L$=""
1160 W$="DO "+V$(1,1)+" = "
1170 X$="$$":Y$=R$(1,I):GOSUB 1300
1180 GOSUB 1350
1190 FOR L=3 TO LO
1200 IF R9=1 AND D(I,L-2)=0 THEN GOTO 1240
1210 W$=V$(1,L):X$="$$":Y$=R$(1,I):GOSUB 1300
1220 GOSUB 1450
1230 GOSUB 1350
1240 NEXT L
1250 W$=";":GOSUB 1350
1260 PRINT L$:PRINT#1,L$:NEXT I
1270 GOSUB 880
1280 PRINT"END OF FILE":PRINT#1,"END OF FILE":CLOSE 1:END
1290 REM ----------SUBROUTINE TO REPLACE STRING X$ WITH STRING Y$ IN STRING W$--
1300 P0=INSTR(W$,X$) 'REPLACE X$ WITH Y$ IN W$
1310 REM ----------SUBROUTINE TO REMOVE SUFFIXES FROM 'DO' EQUATIONS------------
1320 IF P0=0 THEN RETURN
1330 MID$(W$,P0)=Y$:GOTO 1300
1340 REM ----------SUBROUTINE TO PRINT LINE IF NEW ONE > 72 CHARS.--------------
1350 IF LEN(L$)+LEN(W$)<=72 THEN GOTO 1370 'PRINT IF LEN>80
1360 PRINT L$:PRINT#1,L$:L$=""
1370 L$=L$+W$:RETURN
1380 REM ----------SUBROUTINE TO REMOVE STRING X$ FROM STRING W$----------------
1390 P0=INSTR(W$,X$) 'REMOVE STRING X$ FROM W$
1400 IF P0=0 THEN RETURN
1410 IF P1=1 THEN W$=RIGHT$(W$,LEN(W$)-LEN(X$))
1420 IF P0=LEN(W$)-LEN(X$)+1 THEN W$=LEFT$(W$,LEN(W$)-LEN(X$)) ELSE GOTO 1440
1430 GOTO 1390
1440 W$=LEFT$(W$,P0-1)+RIGHT$(W$,LEN(W$)-LEN(X$)-P0+1):GOTO 1390
1450 X$="'POLN":GOSUB 1390
1460 X$="'POLP":GOSUB 1390
1470 X$="'NPOL":GOSUB 1390
1480 X$="'DEF":GOSUB 1390
1490 X$="'NXS":GOSUB 1390
1500 X$="'XNS":GOSUB 1390
1510 X$="'POL":GOSUB 1390
1520 X$="'XN":GOSUB 1390
1530 X$="'XP":GOSUB 1390
1540 X$="'NX":GOSUB 1390
1550 X$="'N":GOSUB 1390
1560 X$="'X":GOSUB 1390
1570 RETURN
1580 REM ----------SUBROUTINE TO CLEAR SCREEN---------------------------------
1590 PRINT CHR$(27);CHR$(58):RETURN
```

The purpose of EQDUPLIC is, given an appropriate input 'template' equation and an input matrix of 0's and 1's, to write out a full set of TROLL equations for all GOL commodities containing the desired terms. Each term in the template equation is associated with a matrix column and will be written out for a particular commodity if the column contains a "1". The program allows this 'thinning' option to be ignored if the user wishes.

The $$'s in the equation template are replaced by commodity codes in the left row of the input matrix. The columns of the input matrix show the terms of the template equation to be included in each commodity equation.

```
USPS.TXT
USPS$$
'DEF ==
ABSV'F(USPD$$'N
-USTC$$'POLN      <-- Input equation template file
-USMD$$'N
-USTP$$'POLN
)
```

Input matrix file of 0's and 1's

```
USPS.PRN(USPD$$'N-USTC$$'P-USMD$$'N-USTP$$'P          )

        BF          1          0          1          0          1
        PK          1          0          1          0          1
        ML          1          0          1          0          1
        DM          1          0          1          0          1
        PM          1          0          1          0          1
        PE          1          0          1          0          1
        WH          1          0          1          0          1
        CN          1          0          1          0          1
        CG          1          0          1          0          1
        RI          1          0          1          0          1
        SB          1          0          1          0          1
        OS          1          0          1          0          1
        SM          1          0          1          0          1
        SO          1          0          1          0          1
        OM          1          0          1          0          1
        OO          1          0          1          0          1
        DB          1          0          1          0          1
        DC          1          0          1          0          1
        DO          1          0          1          0          1
```

Example output file 1

```
BEGINNING OF FILE
---------+---------+---------+---------+---------+---------+---------+---------+
USPSBF: USPSBF'DEF ==ABSV'F(USPDBF'N-USTCBF'POLN-USMDBF'N-USTPBF'POLN),
USPSPK: USPSPK'DEF ==ABSV'F(USPDPK'N-USTCPK'POLN-USMDPK'N-USTPPK'POLN),
USPSML: USPSML'DEF ==ABSV'F(USPDML'N-USTCML'POLN-USMDML'N-USTPML'POLN),
USPSDM: USPSDM'DEF ==ABSV'F(USPDDM'N-USTCDM'POLN-USMDDM'N-USTPDM'POLN),
USPSPM: USPSPM'DEF ==ABSV'F(USPDPM'N-USTCPM'POLN-USMDPM'N-USTPPM'POLN),
USPSPE: USPSPE'DEF ==ABSV'F(USPDPE'N-USTCPE'POLN-USMDPE'N-USTPPE'POLN),
USPSWH: USPSWH'DEF ==ABSV'F(USPDWH'N-USTCWH'POLN-USMDWH'N-USTPWH'POLN),
USPSCN: USPSCN'DEF ==ABSV'F(USPDCN'N-USTCCN'POLN-USMDCN'N-USTPCN'POLN),
USPSCG: USPSCG'DEF ==ABSV'F(USPDCG'N-USTCCG'POLN-USMDCG'N-USTPCG'POLN),
USPSRI: USPSRI'DEF ==ABSV'F(USPDRI'N-USTCRI'POLN-USMDRI'N-USTPRI'POLN),
USPSSB: USPSSB'DEF ==ABSV'F(USPDSB'N-USTCSB'POLN-USMDSB'N-USTPSB'POLN),
USPSOS: USPSOS'DEF ==ABSV'F(USPDOS'N-USTCOS'POLN-USMDOS'N-USTPOS'POLN),
USPSSM: USPSSM'DEF ==ABSV'F(USPDSM'N-USTCSM'POLN-USMDSM'N-USTPSM'POLN),
USPSSO: USPSSO'DEF ==ABSV'F(USPDSO'N-USTCSO'POLN-USMDSO'N-USTPSO'POLN),
USPSOM: USPSOM'DEF ==ABSV'F(USPDOM'N-USTCOM'POLN-USMDOM'N-USTPOM'POLN),
USPSOO: USPSOO'DEF ==ABSV'F(USPDOO'N-USTCOO'POLN-USMDOO'N-USTPOO'POLN),
USPSDB: USPSDB'DEF ==ABSV'F(USPDDB'N-USTCDB'POLN-USMDDB'N-USTPDB'POLN),
USPSDC: USPSDC'DEF ==ABSV'F(USPDDC'N-USTCDC'POLN-USMDDC'N-USTPDC'POLN),
USPSDO: USPSDO'DEF ==ABSV'F(USPDDO'N-USTCDO'POLN-USMDDO'N-USTPDO'POLN),
---------+---------+---------+---------+---------+---------+---------+---------+
DO USPSBF = ABSV'F(USPDBF-USTCBF-USMDBF-USTPBF);
DO USPSPK = ABSV'F(USPDPK-USTCPK-USMDPK-USTPPK);
DO USPSML = ABSV'F(USPDML-USTCML-USMDML-USTPML);
DO USPSDM = ABSV'F(USPDDM-USTCDM-USMDDM-USTPDM);
DO USPSPM = ABSV'F(USPDPM-USTCPM-USMDPM-USTPPM);
DO USPSPE = ABSV'F(USPDPE-USTCPE-USMDPE-USTPPE);
DO USPSWH = ABSV'F(USPDWH-USTCWH-USMDWH-USTPWH);
DO USPSCN = ABSV'F(USPDCN-USTCCN-USMDCN-USTPCN);
DO USPSCG = ABSV'F(USPDCG-USTCCG-USMDCG-USTPCG);
DO USPSRI = ABSV'F(USPDRI-USTCRI-USMDRI-USTPRI);
DO USPSSB = ABSV'F(USPDSB-USTCSB-USMDSB-USTPSB);
DO USPSOS = ABSV'F(USPDOS-USTCOS-USMDOS-USTPOS);
DO USPSSM = ABSV'F(USPDSM-USTCSM-USMDSM-USTPSM);
DO USPSSO = ABSV'F(USPDSO-USTCSO-USMDSO-USTPSO);
DO USPSOM = ABSV'F(USPDOM-USTCOM-USMDOM-USTPOM);
DO USPSOO = ABSV'F(USPDOO-USTCOO-USMDOO-USTPOO);
DO USPSDB = ABSV'F(USPDDB-USTCDB-USMDDB-USTPDB);
DO USPSDC = ABSV'F(USPDDC-USTCDC-USMDDC-USTPDC);
DO USPSDO = ABSV'F(USPDDO-USTCDO-USMDDO-USTPDO);
---------+---------+---------+---------+---------+---------+---------+---------+
END OF FILE
```

TROLL equations

TROLL 'DO' statements to create definition data from historical data

Two examples of output files created from the input files are presented.

The first example did not use the option of 'thinning' out the output equations while the second example does exercise this option. The outputs from this example are the basic ingredients for the domestic price program of TROLL (USGOLPS) presented earlier in this report.

```
BEGINNING OF FILE
---------+---------+---------+---------+---------+---------+---------+---------+
USPSBF: USPSBF'DEF ==ABSV'F(USPDBF'N-USMDBF'N),
USPSPK: USPSPK'DEF ==ABSV'F(USPDPK'N-USMDPK'N),
USPSML: USPSML'DEF ==ABSV'F(USPDML'N-USMDML'N),
USPSDM: USPSDM'DEF ==ABSV'F(USPDDM'N-USMDDM'N),
USPSPM: USPSPM'DEF ==ABSV'F(USPDPM'N-USMDPM'N),
USPSPE: USPSPE'DEF ==ABSV'F(USPDPE'N-USMDPE'N),
USPSWH: USPSWH'DEF ==ABSV'F(USPDWH'N-USMDWH'N),
USPSCN: USPSCN'DEF ==ABSV'F(USPDCN'N-USMDCN'N),
USPSCG: USPSCG'DEF ==ABSV'F(USPDCG'N-USMDCG'N),          Example output file 2
USPSRI: USPSRI'DEF ==ABSV'F(USPDRI'N-USMDRI'N),
USPSSB: USPSSB'DEF ==ABSV'F(USPDSB'N-USMDSB'N),
USPSOS: USPSOS'DEF ==ABSV'F(USPDOS'N-USMDOS'N),
USPSSM: USPSSM'DEF ==ABSV'F(USPDSM'N-USMDSM'N),
USPSSO: USPSSO'DEF ==ABSV'F(USPDSO'N-USMDSO'N),
USPSOM: USPSOM'DEF ==ABSV'F(USPDOM'N-USMDOM'N),
USPSOO: USPSOO'DEF ==ABSV'F(USPDOO'N-USMDOO'N),
USPSDB: USPSDB'DEF ==ABSV'F(USPDDB'N-USMDDB'N),
USPSDC: USPSDC'DEF ==ABSV'F(USPDDC'N-USMDDC'N),
USPSDO: USPSDO'DEF ==ABSV'F(USPDDO'N-USMDDO'N),
---------+---------+---------+---------+---------+---------+---------+---------+
DO USPSBF = ABSV'F(USPDBF-USMDBF);
DO USPSPK = ABSV'F(USPDPK-USMDPK);
DO USPSML = ABSV'F(USPDML-USMDML);
DO USPSDM = ABSV'F(USPDDM-USMDDM);
DO USPSPM = ABSV'F(USPDPM-USMDPM);
DO USPSPE = ABSV'F(USPDPE-USMDPE);
DO USPSWH = ABSV'F(USPDWH-USMDWH);
DO USPSCN = ABSV'F(USPDCN-USMDCN);
DO USPSCG = ABSV'F(USPDCG-USMDCG);
DO USPSRI = ABSV'F(USPDRI-USMDRI);
DO USPSSB = ABSV'F(USPDSB-USMDSB);
DO USPSOS = ABSV'F(USPDOS-USMDOS);
DO USPSSM = ABSV'F(USPDSM-USMDSM);
DO USPSSO = ABSV'F(USPDSO-USMDSO);
DO USPSOM = ABSV'F(USPDOM-USMDOM);
DO USPSOO = ABSV'F(USPDOO-USMDOO);
DO USPSDB = ABSV'F(USPDDB-USMDDB);
DO USPSDC = ABSV'F(USPDDC-USMDDC);
DO USPSDO = ABSV'F(USPDDO-USMDDO);
---------+---------+---------+---------+---------+---------+---------+---------+
END OF FILE
```

II-B.1.c. CTYLINK - Write out TROLL Cross-Country Equations

The linkage of country models requires that equations and other TROLL statements be written using variables for included countries/ regions. The CTYLINK program allows the use of an equation template and a matrix of "1's" (for included countries/regions) as an easy way to write such statements. An example of inputs to, and outputs from this program are given after the listing.

```
10 REM CREATED BY VERNON OLEY RONINGEN - MAR. 20, 1983
20 CLEAR 5000
30 REM THIS PROGRAM REQUIRES 2 INPUT FILES:
40 REM AN EQUATION MASTER FILE (--.TXT) AND A ZERO-ONE MATRIX FILE (--.PRN)
50 REM
60 REM FILE ----.TXT MUST HAVE:
70 REM LINE 1 FILE NAME
80 REM LINE 2 VARIABLE NAME - E.G. USQD$$
90 REM LINE 3 REST OF VARIABLE NAME AND EQUAL SIGN
100 REM      E.G. DEF == ABSV'F
110 REM LINE 4-N (LIMIT N=30) ONE VARIABLE IN EACH LINE
120 REM      E.G. -USQD$$'N
130 REM     REMEMBER TO INCLUDE ALL NECESSARY PARENTHESES WITH
140 REM     WITH VARIABLES OR ON EXTRA LINES
150 REM
160 REM FILE ----.PRN MUST HAVE:
170 REM LINE 1 FILE NAME PLUS DESIRED COLUMN LABELS
180 REM LINE 2 BLANK LINE
190 REM LINE 3-N ROWS FOR EACH GOL COMMODITY WITH A 1 OR 0 FOR EACH TERM OR
200 REM      EACH VARIABLE IN LINES 4-N, A '1' MEANS THAT
210 REM      THE VARIABLE SHOULD BE INCLUDED IN THE EQUATION
220 REM      A '0' MEANS THAT A VARIABLE IS NOT INCLUDED
230 REM      REMEMBER THAT THE FIELDS OF THE COUNTRY MATRIX MUST BE 3 CHAR. WIDE
240 REM
250 REM ----------SAMPLE INPUT FILES FOLLOW:
260 REM ------------------------------------------------------------------------
270 REM LINE 1 - WD.TXT
280 REM LINE 2 - WDQT$$
290 REM LINE 3 - 'DEF ==
300 REM LINE 4 -  ##QT$$
310 REM LINE 5 - +##QT$$
320 REM LINE 6 - +##QT$$
330 REM ------------------------------------------------------------------------
340 REM LINE 1 - WDT US JP RW
350 REM LINE 2 -
360 REM LINE 3 -  BF  1  1  1
370 REM LINE 4 -  CN  1  0  1
380 REM ------------------------------------------------------------------------
390 GOSUB 1620
400 PRINT" C T Y L I N K   BY VOR":PRINT
410 PRINT"PROGRAM TO LINK GOL COUNTRY VARIABALES":PRINT
420 W$="":R9=1:INPUT"DO YOU WANT TO THIN THE EQUATIONS (Y OR N)";W$
430 PRINT:IF W$="N" THEN R9=0
440 INPUT"ENTER NAME OF EQUATION FILE (.TXT FILE)";W$:PRINT
450 REM ----------READ EQUATION MASTER FILE-----------------------------------
460 PRINT"READING EQUATION FILE":PRINT
470 F$="B:"+W$+".TXT":OPEN"I",1,F$
480 DIM V$(1,35):LO=0
490 I8=0
500 LINE INPUT#1,X$
510 LO=LO+1
520 LINE INPUT#1,V$(1,LO)
530 IF EOF(1)THEN GOTO 550
540 PRINT V$(1,LO):GOTO 510
550 PRINT V$(1,LO):CLOSE 1
560 REM ----------READ ZERO-ONE MATRIX FILE-----------------------------------
570 PRINT:F$="B:"+W$+".PRN":PRINT"READING MATRIX FILE":PRINT
580 OPEN"I",1,F$:F$="B:"+W$+"R.TXT"
590 DIM C$(1,35),D(19,35),R$(1,19)
600 LINE INPUT#1,W$:JO=LEN(W$)/3-1
610 FOR J=1 TO JO:U$=MID$(W$,J*3+1,3):GOSUB 790
620 C$(1,J)=U$
630 PRINT"COLUMN LABEL ";J,C$(1,J)
640 NEXT J:I0=0
```

```
650 LINE INPUT#1,W$
660 PRINT
670 LINE INPUT#1,W$:I0=I0+1
680 IF EOF(1) THEN CLOSE 1:I8=1
690 U$=LEFT$(W$,3):GOSUB 790
700 R$(1,I0)=U$
710 PRINT"ROW (COMMODITY) LABEL ";I0,R$(1,I0)
720 FOR J=1 TO J0
730 U$=MID$(W$,J*3+1,3)
740 GOSUB 790
750 D(I0,J)=VAL(U$)
760 NEXT J
770 IF I8 = 0 THEN GOTO 670 ELSE GOTO 950
780 REM ----------SUBROUTINE TO REMOVE LEADING AND TRAILING BLANKS FROM Z$----------
790 Z$=U$
800 IF LEFT$(Z$,1)=" "THEN Z$=RIGHT$(Z$,LEN(Z$)-1) ELSE 820
810 GOTO 800
820 IF RIGHT$(Z$,1)=" "THEN Z$=LEFT$(Z$,LEN(Z$)-1) ELSE 840
830 GOTO 820
840 U$=Z$:RETURN
850 REM ----------SUBROUTINE TO REMOVE LEADING BLANKS FROM W$----------
860 'F LEFT$(W$,1)=" "THEN GOTO 870 ELSE GOTO 880
870 W$=RIGHT$(W$,LEN(W$)-1):GOTO 860
880 RETURN
890 REM ----------SUBROUTINE TO PRINT LINES ----------
900 PRINT"---------+---------+---------+---------+---------+---------+"
910 PRINT#1,"---------+---------+---------+---------+---------+---------+---------+---------+"
920 RETURN
930 GOSUB 1620
940 REM ----------WRITE TROLL EQUATIONS----------
950 GOSUB 1620: PRINT:OPEN"O",1,F$:PRINT"WRITE EQUATIONS":PRINT:J5=0
960 FOR I=1 TO I0:IF R$(1,I)=C$(1,I) THEN J5=J5+1:NEXT I
970 PRINT"BEGINNING OF FILE":PRINT#1,"BEGINNING OF FILE"
980 GOSUB 900
990 FOR I=1 TO I0:L$=""
1000 W$=V$(1,1)+": "+V$(1,1)+V$(1,2)
1010 X$="$$":Y$=R$(1,I):GOSUB 1340
1020 GOSUB 1380
1030 FOR L=3 TO L0
1040 IF R9=1 AND D(I,L-2)=0 THEN GOTO 1080
1050 W$=V$(1,L):X$="$$":Y$=R$(1,I):GOSUB 1340
1060 X$="##":Y$=C$(1,L-2):GOSUB 1340
1070 GOSUB 1380
1080 NEXT L
1090 W$=",":GOSUB 1380
1100 PRINT L$:PRINT#1,L$:NEXT I
1110 GOSUB 1620
1120 W$="":PRINT:INPUT"DO YOU WANT DO EQUATIONS AS WELL (Y OR N)";W$
1130 IF W$="N" THEN GOTO 1320
1140 REM ----------WRITE TROLL 'DO' EQUATIONS----------
1150 PRINT:PRINT"WRITE 'DO' EQUATIONS":PRINT
1160 GOSUB 900
1170 REM
1180 FOR I=1 TO I0:L$=""
1190 W$="DO "+V$(1,1)+" = "
1200 X$="$$":Y$=R$(1,I):GOSUB 1340
1210 GOSUB 1380
1220 FOR L=3 TO L0
1230 IF R9=1 AND D(I,L-2)=0 THEN GOTO 1280
1240 W$=V$(1,L):X$="$$":Y$=R$(1,I):GOSUB 1340
1250 X$="##":Y$=C$(1,L-2):GOSUB 1340
1260 GOSUB 1480
1270 GOSUB 1380
1280 NEXT L
1290 W$=";":GOSUB 1380
1300 PRINT L$:PRINT#1,L$:NEXT I
1310 GOSUB 900
1320 PRINT"END OF FILE":PRINT#1,"END OF FILE":CLOSE 1:END
1330 REM ----------SUBROUTINE TO REPLACE STRING X$ WITH STRING Y$ IN STRING W$--
1340 P0=INSTR(W$,X$) 'REPLACE X$ WITH Y$ IN W$
1350 IF P0=0 THEN RETURN
1360 MID$(W$,P0)=Y$:GOTO 1340
```

```
1370 REM ----------SUBROUTINE TO PRINT LINE IF NEW ONE > 72 CHARS.----------
1380 IF LEN(L$)+LEN(W$)<=72 THEN GOTO 1400 'PRINT IF LEN>80
1390 PRINT L$:PRINT#1,L$:L$=""
1400 L$=L$+W$:RETURN
1410 REM ----------SUBROUTINE TO REMOVE STRING X$ FROM STRING W$----------
1420 P0=INSTR(W$,X$) 'REMOVE STRING X$ FROM W$
1430 IF P0=0 THEN RETURN
1440 IF P1=1 THEN W$=RIGHT$(W$,LEN(W$)-LEN(X$))
1450 IF P0=LEN(W$)-LEN(X$)+1 THEN W$=LEFT$(W$,LEN(W$)-LEN(X$)) ELSE GOTO 1470
1460 GOTO 1420
1470 W$=LEFT$(W$,P0-1)+RIGHT$(W$,LEN(W$)-LEN(X$)-P0+1):GOTO 1420
1480 X$="'POLN":GOSUB 1420
1490 X$="'POLP":GOSUB 1420
1500 X$="'NPOL":GOSUB 1420
1510 X$="'DEF":GOSUB 1420
1520 X$="'NXS":GOSUB 1420
1530 X$="'XNS":GOSUB 1420
1540 X$="'POL":GOSUB 1420
1550 X$="'XN":GOSUB 1420
1560 X$="'XP":GOSUB 1420
1570 X$="'NX":GOSUB 1420
1580 X$="'N":GOSUB 1420
1590 X$="'X":GOSUB 1420
1600 RETURN
1610 REM ----------SUBROUTINE TO CLEAR SCREEN----------
1620 PRINT CHR$(27);CHR$(58):RETURN
```

Input equation template file

```
WDT.TXT          +##QT$$
WDQT$$           +##QT$$
'DEF ==          +##QT$$
 ##QT$$          +##QT$$
+##QT$$          +##QT$$
+##QT$$          +##QT$$
+##QT$$          +##QT$$
+##QT$$          +##QT$$
+##QT$$          +##QT$$
+##QT$$          +##QT$$
+##QT$$          +##QT$$
+##QT$$          +##QT$$
+##QT$$          +##QT$$
+##QT$$          +##QT$$
+##QT$$          +RWQT$$
```

The purpose of the program CTYLINK is, given an input 'template' equation and a matrix of 1's and _'s, to write out the template equations for the countries/ regions marked at the heads of the columns. If the column has a "1" for a particular country/region, the term will be included for that country/region for the row commodity.

SUPERCALC ".CAL" file used to create the input country matrix file which is read by the program.

```
  |A||B||C||D||E||F||G||H||I||J||K||L||M||N||O||P||Q||R||S||T||U||V||W||X||Y||Z|
 1|WDT US CN EC WE JP AZ SF EE SV CH MX CA BZ AR VE LA AF EG ME ND OS DO TH SA EA OA
 2|
 3| BF  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _
 4| PK  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _
 5| ML  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _
 6| PM  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _
 7| PE  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _
 8| WH  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _
 9| CN  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _
10| CG  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _
11| RI  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _
12| SB  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _
13| OS  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _
14| SM  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _
15| SO  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _
16| OM  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _
17| OO  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _
18| DB  1  1  1  _  0  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _
19| DC  1  1  1  _  0  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _
20| DO  1  1  1  _  0  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _
```

Input country matrix of 1's and -'s

```
WDT US CN EC WE JP AZ SF EE SV CH MX CA BZ AR VE LA AF EG ME ND OS DO TH SA EA OA RW

BF  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  1
PK  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  1
ML  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  1
PM  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  1
PE  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  1
WH  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  1
CN  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  1
CG  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  1
RI  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  1
SB  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  1
OS  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  1
SM  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  1
SO  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  1
OM  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  1
OO  1  1  1  _  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  1
DB  1  1  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  1
DC  1  1  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  1
DO  1  1  1  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  1
```

Example output file

```
BEGINNING OF FILE
---------+---------+---------+---------+---------+---------+---------+---------+
WDQTBF: WDQTBF'DEF == USQTBF+CNQTBF+ECQTBF+JPQTBF+RWQTBF,
WDQTPK: WDQTPK'DEF == USQTPK+CNQTPK+ECQTPK+JPQTPK+RWQTPK,
WDQTML: WDQTML'DEF == USQTML+CNQTML+ECQTML+JPQTML+RWQTML,
WDQTPM: WDQTPM'DEF == USQTPM+CNQTPM+ECQTPM+JPQTPM+RWQTPM,
WDQTPE: WDQTPE'DEF == USQTPE+CNQTPE+ECQTPE+JPQTPE+RWQTPE,
WDQTWH: WDQTWH'DEF == USQTWH+CNQTWH+ECQTWH+JPQTWH+RWQTWH,
WDQTCN: WDQTCN'DEF == USQTCN+CNQTCN+ECQTCN+JPQTCN+RWQTCN,
WDQTCG: WDQTCG'DEF == USQTCG+CNQTCG+ECQTCG+JPQTCG+RWQTCG,
WDQTRI: WDQTRI'DEF == USQTRI+CNQTRI+ECQTRI+JPQTRI+RWQTRI,    ← TROLL equations
WDQTSB: WDQTSB'DEF == USQTSB+CNQTSB+ECQTSB+JPQTSB+RWQTSB,
WDQTOS: WDQTOS'DEF == USQTOS+CNQTOS+ECQTOS+JPQTOS+RWQTOS,
WDQTSM: WDQTSM'DEF == USQTSM+CNQTSM+ECQTSM+JPQTSM+RWQTSM,
WDQTSO: WDQTSO'DEF == USQTSO+CNQTSO+ECQTSO+JPQTSO+RWQTSO,
WDQTOM: WDQTOM'DEF == USQTOM+CNQTOM+ECQTOM+JPQTOM+RWQTOM,
WDQTOO: WDQTOO'DEF == USQTOO+CNQTOO+ECQTOO+JPQTOO+RWQTOO,
WDQTDB: WDQTDB'DEF == USQTDB+CNQTDB+ECQTDB+RWQTDB,
WDQTDC: WDQTDC'DEF == USQTDC+CNQTDC+ECQTDC+RWQTDC,
WDQTDO: WDQTDO'DEF == USQTDO+CNQTDO+ECQTDO+RWQTDO,
---------+---------+---------+---------+---------+---------+---------+---------+
DO WDQTBF =  USQTBF+CNQTBF+ECQTBF+JPQTBF+RWQTBF;
DO WDQTPK =  USQTPK+CNQTPK+ECQTPK+JPQTPK+RWQTPK;
DO WDQTML =  USQTML+CNQTML+ECQTML+JPQTML+RWQTML;
DO WDQTPM =  USQTPM+CNQTPM+ECQTPM+JPQTPM+RWQTPM;
DO WDQTPE =  USQTPE+CNQTPE+ECQTPE+JPQTPE+RWQTPE;
DO WDQTWH =  USQTWH+CNQTWH+ECQTWH+JPQTWH+RWQTWH;
DO WDQTCN =  USQTCN+CNQTCN+ECQTCN+JPQTCN+RWQTCN;
DO WDQTCG =  USQTCG+CNQTCG+ECQTCG+JPQTCG+RWQTCG;
DO WDQTRI =  USQTRI+CNQTRI+ECQTRI+JPQTRI+RWQTRI;    ← TROLL 'DO' equations
DO WDQTSB =  USQTSB+CNQTSB+ECQTSB+JPQTSB+RWQTSB;
DO WDQTOS =  USQTOS+CNQTOS+ECQTOS+JPQTOS+RWQTOS;
DO WDQTSM =  USQTSM+CNQTSM+ECQTSM+JPQTSM+RWQTSM;
DO WDQTSO =  USQTSO+CNQTSO+ECQTSO+JPQTSO+RWQTSO;
DO WDQTOM =  USQTOM+CNQTOM+ECQTOM+JPQTOM+RWQTOM;
DO WDQTOO =  USQTOO+CNQTOO+ECQTOO+JPQTOO+RWQTOO;
DO WDQTDB =  USQTDB+CNQTDB+ECQTDB+RWQTDB;
DO WDQTDC =  USQTDC+CNQTDC+ECQTDC+RWQTDC;
DO WDQTDO =  USQTDO+CNQTDO+ECQTDO+RWQTDO;
---------+---------+---------+---------+---------+---------+---------+---------+
END OF FILE
```

II-B.1. d. CLONE - Clone a TROLL Statement for GOL Commodities

The program CLONE takes an input 'template' TROLL equation or statement and replaces the $$'s with GOL commodity codes thus cloning the statement for all GOL commodities. The user has the option of omitting selected commodities from the cloning process.

```
10 REM CREATED BY VERNON OLEY RONINGEN - FEB. 8, 1983
20 PRINT CHR$(27):CHR$(58):CLEAR 5000
30 DIM A$(1,19),D$(1,30)
40 PRINT "C L O N E  P R O G R A M  BY VOR":PRINT
50 PRINT:PRINT"THIS PROGRAM DUPLICATES TROLL COMMANDS OR EQUATIONS FROM A "
60 PRINT"MASTER SET CREATED BY A WORD PROCESSING PROGRAM AND STORED IN ASCII FORMAT"
70 PRINT"EACH '$$' IS REPLACED BY GOL COMMODITY CODES":PRINT
80 PRINT"PROGRAM REQUIRES AN ASCII FILE OF NO MORE THAN 31 LINES ON 'B' DRIVE
90 PRINT"THE NAME MUST BE *******.TXT"
100 PRINT"THE FIRST LINE CAN CONTAIN THE FILE NAME AND IS NOT CLONED"
110 PRINT"A SAMPLE FILE IS:"
120 PRINT
130 PRINT"FILE NAME"
140 PRINT"1ST LINE TO BE CLONED"
150 PRINT"2ND LINE TO BE CLONED"
160 PRINT"ETC.":PRINT
170 PRINT"FETCHING COMMODITY CODES"
180 GOSUB 440 ' FETCH COMMODITY CODES
190 GOSUB 500 ' READ IN FILE TO BE CLONED
200 GOSUB 630 ' SELECT PRODUCTS TO BE OMITTED
210 GOSUB 290 ' WRITE FILE OF CLONED STATEMENTS
220 END
230 REM
240 REM ----------SUBROUTINE TO REPLACE '$$' WITH A GOL COMMODITY CODE---------
250 P0=INSTR(W$,"$$") 'REPLACE $$ WITH GOL COMMODITY CODE
260 IF P0=0 THEN RETURN
270 MID$(W$,P0)=A$(1,I):GOTO 250
280 REM ----------SUBROUTINE TO WRITE A FILE OF CLONED STATEMENTS--------------
290 PRINT CHR$(27);CHR$(58):PRINT"WRITING FILE OF CLONED STATEMENTS":PRINT
300 F$="B:"+F$+"C.TXT:1":OPEN"O",1,F$
310 PRINT"BEGINNING OF FILE":PRINT#1,"BEGINNING OF FILE"
320 FOR I=1 TO 19
330 IF O(I)=1 THEN 400
340 FOR J=1 TO 10
350 W$=D$(1,J)
360 GOSUB 250
370 PRINT W$
380 PRINT#1,W$
390 NEXT J
400 NEXT I
410 PRINT"END OF FILE":PRINT#1,"END OF FILE"
420 RETURN
```

```
430 REM ----------SUBROUTINE TO PUT GOL COMMODITY CODES IN A VECTOR------------
440 W$="BFPKMLDMPMPEWHCNCGRISBOSSMSOOMOODBDCDO" 'GOL COMMODITY CODES CONCATENATED
450 PRINT:FOR I=1 TO 19
460 A$(1,I)=MID$(W$,(I-1)*2+1,2)
470 PRINT MID$(W$,(I-1)*2+1,2);" ";
480 NEXT I:PRINT " "
490 PRINT:RETURN
500 REM ----------SUBROUTINE TO READ IN AN ASCII FILE TO BE CLONED-------------
510 PRINT:INPUT"ENTER NAME OF 'TXT' FILE ON DRIVE 'B' TO BE CLONED";F$
520 W$="B:"+F$+".TXT":OPEN"I",1,W$
530 PRINT CHR$(27);CHR$(58)
540 PRINT "FILE READ IN IS:":PRINT
550 PRINT W$
560 LINE INPUT#1,W$
570 I=0
580 I=I+1:LINE INPUT#1,D$(1,I)
590 IF EOF(1) THEN 610
600 PRINT D$(1,I):GOTO 580
610 I0=I:CLOSE 1:PRINT D$(1,I0):PRINT:RETURN
620 REM ----------SUBROUTINE TO OMIT GOL PRODUCTS FROM CLONING PROCEDURE-------
630 PRINT"SELECT PRODUCTS TO OMIT FROM CLONING"
640 DIM O(19):FOR I=1 TO 19:O(I)=0:NEXT I
650 FOR I=1 TO 19:PRINT I;A$(1,I),:NEXT I:PRINT:PRINT
660 O1=0:INPUT"SELECT NUMBER OF PRODUCT TO OMIT (PRESS 'ENTER' TO RETURN)";O1
670 IF O1=0 THEN RETURN
680 IF O1<1 OR O1>19 THEN 660 ELSE O(O1)=1:GOTO 660
```

TEST.TXT

```
DO USQT$$ = USQS$$ - USQD$$ ;
OLS USQD$$ USPD$$ USINC ;
```

Input Template file of TROLL statements

Output file of 'cloned' TROLL statements

```
BEGINNING OF FILE
DO USQTBF = USQSBF - USQDBF ;
OLS USQDBF USPDBF USINC ;
DO USQTPK = USQSPK - USQDPK ;
OLS USQDPK USPDPK USINC ;
DO USQTML = USQSML - USQDML ;
OLS USQDML USPDML USINC ;
END OF FILE
```

II-B.1.e. COMCLONE - Clone a TROLL Comment for GOL Commodities

COMCLONE takes a file of representative TROLL comment statements and replaces the $$'s with selected commodity codes and the ##'s with the commodity description associated with those codes.

```
10 REM CREATED BY VERNON OLEY RONINGEN - FEB. 8, 1983
20 PRINT CHR$(27);CHR$(58):CLEAR 5000
30 DIM A$(1,19),D$(1,30),P$(1,19)
40 PRINT "C O M C L O N E  P R O G R A M  BY VOR":PRINT
50 PRINT:PRINT"THIS PROGRAM DUPLICATES TROLL COMMENTS FROM A "
60 PRINT"MASTER SET CREATED BY A WORD PROCESSING PROGRAM AND STORED IN ASCII FORMAT"
70 PRINT"EACH '$$' IS REPLACED BY A GOL COMMODITY CODES"
80 PRINT"EACH '##' IS REPLACE BY A GOL COMMODITY NAME":PRINT
90 PRINT"PROGRAM REQUIRES AN ASCII FILE OF NO MORE THAN 31 LINES ON 'B' DRIVE
100 PRINT"THE NAME MUST BE *******.TXT"
110 PRINT"THE FIRST LINE CAN CONTAIN THE FILE NAME AND IS NOT CLONED"
120 PRINT"A SAMPLE FILE IS:"
130 PRINT
140 PRINT"FILE NAME"
150 PRINT"1ST LINE TO BE CLONED"
160 PRINT"2ND LINE TO BE CLONED"
170 PRINT"ETC.":PRINT
180 PRINT"FETCHING COMMODITY CODES"
190 GOSUB 510 ' FETCH COMMODITY CODES
200 PRINT:PRINT"FETCHING COMMODITY NAMES"
210 GOSUB 760 'FETCH COMMODITY NAMES
220 GOSUB 570 ' READ IN FILE TO BE CLONED
230 GOSUB 700 ' SELECT PRODUCTS TO BE OMITTED
240 GOSUB 350 ' WRITE FILE OF CLONED STATEMENTS
250 END
260 REM
270 REM ----------SUBROUTINE TO REPLACE '$$' WITH A GOL COMMODITY CODE---------
280 PO=INSTR(W$,"$$") 'REPLACE $$ WITH GOL COMMODITY CODE
290 IF PO=0 THEN RETURN
300 MID$(W$,PO)=A$(1,I):GOTO 280
310 REM ----------SUBROUTINE TO REPLACE '##' WITH GOL COMMODITY NAMES
320 PO=INSTR(W$,"##"):IF PO=0 THEN RETURN
330 W$=LEFT$(W$,PO-1)+P$(1,I)+RIGHT$(W$,LEN(W$)-PO-1):RETURN
340 REM ----------SUBROUTINE TO WRITE A FILE OF CLONED STATEMENTS--------------
350 PRINT CHR$(27);CHR$(58):PRINT"WRITING FILE OF CLONED STATEMENTS":PRINT
360 F$="B:"+F$+"C.TXT:1":OPEN"O",1,F$
370 PRINT"BEGINNING OF FILE":PRINT#1,"BEGINNING OF FILE"
380 FOR J=1 TO 10
390 FOR I=1 TO 19
400 IF O(I)=1 THEN GOTO 460
410 W$=D$(1,J)
420 GOSUB 280
430 GOSUB 310
440 PRINT W$
450 PRINT#1,W$
460 NEXT I
470 NEXT J
480 PRINT"END OF FILE":PRINT#1,"END OF FILE"
490 RETURN
500 REM ----------SUBROUTINE TO PUT GOL COMMODITY CODES IN A VECTOR------------
510 W$="BFPKMLDMPMPEWHCNCGRISBOSSMSOOMOODBDCDO" 'GOL COMMODITY CODES CONCATENATED
520 PRINT:FOR I=1 TO 19
530 A$(1,I)=MID$(W$,(I-1)*2+1,2)
540 PRINT MID$(W$,(I-1)*2+1,2);" ";
550 NEXT I:PRINT " "
560 PRINT:RETURN
```

```
570 REM ----------SUBROUTINE TO READ IN AN ASCII FILE TO BE CLONED-------------
580 PRINT:INPUT"ENTER NAME OF 'TXT' COMMENT FILE ON DRIVE 'B' TO BE CLONED";F$
590 W$="B:"+F$+".TXT":OPEN"I",1,W$
600 PRINT CHR$(27);CHR$(58)
610 PRINT "FILE READ IN IS:":PRINT
620 PRINT W$
630 LINE INPUT#1,W$
640 I=0
650 I=I+1:LINE INPUT#1,D$(1,I)
660 IF EOF(1) THEN 680
670 PRINT D$(1,I):GOTO 650
680 IO=I:CLOSE 1:PRINT D$(1,IO):PRINT:RETURN
690 REM ----------SUBROUTINE TO OMIT GOL PRODUCTS FROM CLONING PROCEDURE-------
700 PRINT"SELECT PRODUCTS TO OMIT FROM CLONING"
710 DIM O(19):FOR I=1 TO 19:O(I)=0:NEXT I
720 FOR I=1 TO 19:PRINT I;A$(1,I),:NEXT I:PRINT:PRINT
730 O1=0:INPUT"SELECT NUMBER OF PRODUCT TO OMIT (PRESS 'ENTER' TO RETURN)";O1
740 IF O1=0 THEN RETURN
750 IF O1<1 OR O1>19 THEN 730 ELSE O(O1)=1:GOTO 730
760 REM ----------SUBROUTINE TO PUT GOL COMMODITY NAMES IN A VECTOR
770 P$(1,1)="BEEF+VEAL"
780 P$(1,2)="PORK"
790 P$(1,3)="MUTTON+LAMB"
800 P$(1,4)="DAIRY-MILK"
810 P$(1,5)="POULTRY-MEAT"
820 P$(1,6)="POULTRY-EGGS"
830 P$(1,7)="WHEAT"
840 P$(1,8)="CORN"
850 P$(1,9)="OTHER COARSE GRAINS"
860 P$(1,10)="RICE"
870 P$(1,11)="SOYBEANS"
880 P$(1,12)="OTHER OILSEEDS"
890 P$(1,13)="SOYMEAL"
900 P$(1,14)="SOYOIL"
910 P$(1,15)="OTHER MEALS"
920 P$(1,16)="OTHER OILS"
930 P$(1,17)="DAIRY-BUTTER"
940 P$(1,18)="DAIRY-CHEESE"
950 P$(1,19)="DAIRY-OTHER PRODUCTS"
960 PRINT:FOR I=1 TO 19
970 PRINT P$(1,I),:NEXT I:PRINT
980 RETURN
```

Input template of TROLL comment statements

```
QTCOM.TXT
SYMCOM USQT$$ QUANTITY TRADED * UNITED STATES * ##;
SYMCOM CNQT$$ QUANTITY TRADED * CANADA * ##;
SYMCOM ECQT$$ QUANTITY TRADED * EUROPEAN COMMUNITY * ##;
SYMCOM JPQT$$ QUANTITY TRADED * JAPAN * ##;
SYMCOM RWQT$$ QUANTITY TRADED * REST-OF-WORLD * ##;
```

Output file of 'cloned' TROLL comment statements

```
BEGINNING OF FILE
SYMCOM USQTBF QUANTITY TRADED * UNITED STATES * BEEF+VEAL;
SYMCOM USQTPK QUANTITY TRADED * UNITED STATES * PORK;
SYMCOM USQTML QUANTITY TRADED * UNITED STATES * MUTTON+LAMB;
SYMCOM CNQTBF QUANTITY TRADED * CANADA * BEEF+VEAL;
SYMCOM CNQTPK QUANTITY TRADED * CANADA * PORK;
SYMCOM CNQTML QUANTITY TRADED * CANADA * MUTTON+LAMB;
SYMCOM ECQTBF QUANTITY TRADED * EUROPEAN COMMUNITY * BEEF+VEAL;
SYMCOM ECQTPK QUANTITY TRADED * EUROPEAN COMMUNITY * PORK;
SYMCOM ECQTML QUANTITY TRADED * EUROPEAN COMMUNITY * MUTTON+LAMB;
SYMCOM JPQTBF QUANTITY TRADED * JAPAN * BEEF+VEAL;
SYMCOM JPQTPK QUANTITY TRADED * JAPAN * PORK;
SYMCOM JPQTML QUANTITY TRADED * JAPAN * MUTTON+LAMB;
SYMCOM RWQTBF QUANTITY TRADED * REST-OF-WORLD * BEEF+VEAL;
SYMCOM RWQTPK QUANTITY TRADED * REST-OF-WORLD * PORK;
SYMCOM RWQTML QUANTITY TRADED * REST-OF-WORLD * MUTTON+LAMB;
```

II-B.1. f. MERGE - Merge Files for Transmission to a Mainframe Computer

MERGE combines selected files into one large file. This is useful, for example, if a set of files has to be transmitted to a mainframe computer; one large file can be transmitted rather than a series of small ones.

```
10 REM CREATED BY VERNON OLEY RONINGEN - FEB. 17, 1983
20 PRINT CHR$(27);CHR$(58)
30 PRINT" M E R G E   BY VOR":PRINT
40 PRINT"PROGRAM TO MERGE '.TXT' FILES INTO ONE FILE (T.TXT) FOR TRANSMISSION"
50 PRINT
60 OPEN"O",1,"B:T.TXT"
70 PRINT:PRINT"ROUTINE TO ADD A NEW FILE TO 'T' (HIT RETURN WHEN DONE)"
80 INPUT"ENTER NAME (WITHOUT '.TXT') OF FILE ON DRIVE B TO BE MERGED";N$
90 IF N$="" THEN GOTO 150
100 F$="B:"+N$+".TXT":OPEN"I",2,F$
110 LINE INPUT #2,W$
120 IF EOF(2) THEN GOTO 140
130 PRINT #1,W$:PRINT W$:GOTO 110
140 PRINT #1,W$:PRINT W$:CLOSE 2:GOTO 70
150 CLOSE 1:PRINT:PRINT"T.TXT IS DONE":END
```

II-B.1.g. CMATRIX - Create TROLL Commands to Produce a Display Matrix

CMATRIX creates most of the TROLL statements needed to build display matrices for GOL constants and elasticities. Information from an input file is used to build the display matrix.

```
10 REM CREATED BY VERNON OLEY RONINGEN - FEB 15, 1983
20 CLEAR 3000
30 GOSUB 860
40 PRINT" C M A T R I X   BY VOR":PRINT
50 PRINT"PROGRAM TO CREATE A MATRIX TO HOLD CONSTANTS/ELASTICITIES":PRINT
60 GOSUB 1000 'FETCHING GOL COMMODITY CODES
70 REM PROGRAM REQURES AN INPUT (**.TXT) FILE ON DRIVE B AS FOLLOWS:"
80 REM LINE 1 - NAME OF FILE (AND MATRIX)
90 REM LINE 2 - TITLE FOR MATRIX
100 REM LINE 3 - NO. OF COLUMNS IN MATRIX
110 REM LINE 4 - NAME OF CONSTANT IN 1ST COLUMN OF MATRIX
120 REM LINE 5 - TITLE OF 1ST COLUMN IN MATRIX
130 REM LINE 6 - BEGINNING VALUE OF CONSTANT IN 1ST COLUMN
140 REM LINES 7 - 9 REPEAT LINES 4 - 6 FOR 2ND COLUMN IN MATRIX, ETC.
150 REM
160 INPUT"ENTER NAME OF INPUT FILE (WITHOUT **.TXT) ON DRIVE B";N$
170 F$="B:"+N$+".TXT":OPEN"I",1,F$
180 DIM D$(1,60)
190 I0=1
200 REM ----------READ INPUT FILE
210 LINE INPUT #1,D$(1,I0)
220 IF EOF(1) THEN GOTO 240
230 I0=I0+1:GOTO 210
240 CLOSE 1:I0=I0-1:GOSUB 860
250 REM ----------WRITE OUTPUT FILE
260 F$="B:"+N$+"C.TXT":OPEN"O",2,F$
270 W$="BEGINNING OF FILE":GOSUB 880
280 GOSUB 970
```

```
290 W$="OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7;":GOSUB 880
300 W$="&ERROR &IGNORE":GOSUB 880
310 PO=INSTR(D$(1,1),".")-1:M$=LEFT$(D$(1,1),PO)
320 W$="DELETE DATA "+M$+";":GOSUB 880
330 W$="DO "+M$+" = "+"CRMAT(18,"+D$(1,3)+", NA);":GOSUB 880
340 W$="DELETE GENERAL LABEL_"+M$+"C;":GOSUB 880
350 W$="LEDIT "+M$+"C;":GOSUB 880
360 W$="ADD TOP,":GOSUB 880
370 JO=VAL(D$(1,3))
380 FOR J=1 TO JO
390 W$=D$(1,(J-1)*3+5)+",":GOSUB 880
400 NEXT J
410 W$=";FILE;":GOSUB 880
420 W$="DEDIT "+M$+";":GOSUB 880
430 W$="COMMENT "+D$(1,2)+";":GOSUB 880
440 W$="FILE;":GOSUB 880
450 W$="DELETE GENERAL LABEL_"+M$+"T;":GOSUB 880
460 W$="LEDIT "+M$+"T;":GOSUB 880
470 W$="ADD TOP,":GOSUB 880
480 W$=D$(1,2)+";":GOSUB 880
490 W$="FILE;":GOSUB 880
500 GOSUB 900
510 W$="CEDIT "+LEFT$(D$(1,1),2)+"GOL;":GOSUB 880
520 W$="ADD ":GOSUB 880
530 FOR J=1 TO JO
540 FOR I=1 TO 18
550 PO=INSTR(D$(1,(J-1)*3+4),"'")-1
560 W$=LEFT$(D$(1,(J-1)*3+4),PO)+" "+D$(1,(J-1)*3+6)+" ,"
570 GOSUB 920
580 GOSUB 880
590 NEXT I,J
600 W$=";FILE;":GOSUB 880
610 GOSUB 970
620 W$="OUTPUT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;":GOSUB 880
630 W$="CEDIT "+LEFT$(D$(1,1),2)+"GOL;":GOSUB 880
640 W$="CHANGE":GOSUB 880
650 FOR J=1 TO JO
660 FOR I=1 TO 18
670 PO=INSTR(D$(1,(J-1)*3+4),"'")-1
680 W$=LEFT$(D$(1,(J-1)*3+4),PO)+" "+D$(1,(J-1)*3+6)+" ,"
690 GOSUB 920
700 GOSUB 880
710 NEXT I,J
720 W$=";FILE;":GOSUB 880
730 W$="DORANGE;":GOSUB 880
740 W$="BINDVAL CONST "+LEFT$(D$(1,1),2)+"GOL;":GOSUB 880
750 FOR J= 1 TO JO
760 FOR I= 1 TO 18
770 W$="DOCORE "+M$+"=MATREP("+M$+","+D$(1,(J-1)*3+4)+","+STR$(I)+","+STR$(J)+");"
780 GOSUB 920:GOSUB 880
790 NEXT I,J
800 W$="DO "+M$+" = "+M$+";":GOSUB 880
810 GOSUB 900:GOSUB 970
820 W$="END OF FILE":GOSUB 870
830 CLOSE 2:PRINT:PRINT"FILE ";F$;" IS DONE"
840 END
850 REM ----------SUBROUTINE TO CLEAR SCREEN
860 PRINT CHR$(27);CHR$(58):RETURN
870 REM ----------SUBROUTINE TO PRINT A LINE
880 PRINT W$:PRINT #2,W$:RETURN
890 REM ----------SUBROUTINE TO PRINT MATRIX WRITING COMMAND
900 W$="DOCORE OPRTMAT("+M$+",WDROW'L,"+M$+"C'L,0,0,-28,"+M$+"T'L,1);"
910 GOSUB 880:RETURN
920 REM ----------SUBROUTINE TO REPLACE '$$' WITH A GOL COMMODITY CODE
930 PO=INSTR(W$,"$$")
940 IF PO=0 THEN RETURN
950 MID$(W$,PO)=A$(1,I):GOTO 930
960 REM ----------SUBROUTINE TO PRINT LINES
970 PRINT"---------+---------+---------+---------+---------+---------+---------+---------+--"
980 PRINT #2, "---------+---------+---------+---------+---------+---------+---------+---------+--"
990 RETURN
1000 REM ----------SUBROUTINE TO PUT GOL COMMODITY CODES IN A VECTOR
1010 W$="BFPKMLPMPEWHCNCGRISBOSSMSOOMOODBDCDO"
1020 DIM A$(1,18)
1030 PRINT:FOR I=1 TO 18
1040 A$(1,I)=MID$(W$,(I-1)*2+1,2)
1050 PRINT A$(1,I);" ";:NEXT I:PRINT
1060 PRINT:RETURN
```

Input file

```
WDPAR.TXT
PARAMETERS FOR WORLD MARKET CLEARING MECHANISM - WDGOL ← Title of the matrix
 2  ← Number of columns needed
WDCP$$'P  ← Constant for 1st column
CONVERGENCE PARAMETER FOR WORLD PRICE ESTIMATE (0<PC<1) ← Title for the 1st colu
0.05  ← Beginning elasticities in the first column
WDCB$$'P
CONVERGENCE BOUND FOR WORLD PRICE ESTIMATE (0<PB<1)
0.10
```

Example output file

```
BEGINNING OF FILE
---------+---------+---------+---------+---------+---------+---------+--
OUTOPT RMARG 132 FPTFIELD 14 TABWIDTH 7;
&ERROR &IGNORE
DELETE DATA WDPAR;
DO WDPAR = CRMAT(18, 2, NA);  ← Create matrix
DELETE GENERAL LABEL_WDPARC;
LEDIT WDPARC;  ← Create matrix labels and titles
ADD TOP,
CONVERGENCE PARAMETER FOR WORLD PRICE ESTIMATE (0<PC<1),
CONVERGENCE BOUND FOR WORLD PRICE ESTIMATE (0<PB<1),
;FILE;
DEDIT WDPAR;
COMMENT PARAMETERS FOR WORLD MARKET CLEARING MECHANISM - WDGOL:
FILE;
DELETE GENERAL LABEL_WDPART;
LEDIT WDPART;
ADD TOP,
PARAMETERS FOR WORLD MARKET CLEARING MECHANISM - WDGOL;
FILE;
DOCORE OPRTMAT(WDPAR,WDROW'L,WDPARC'L,0,0,-28,WDPART'L,1);
CEDIT WDGOL;
ADD
WDCPBF 0.05 ,
WDCPPK 0.05 ,
WDCPML 0.05 ,
WDCPPM 0.05 ,
WDCPPE 0.05 ,
WDCPWH 0.05 ,
WDCPCN 0.05 ,
WDCPCG 0.05 ,
WDCPRI 0.05 ,
WDCPSB 0.05 ,
WDCPOS 0.05 ,
WDCPSM 0.05 ,
WDCPSO 0.05 ,  ← Create initial elasticities
WDCPOM 0.05 ,
WDCPOO 0.05 ,
WDCPDB 0.05 ,
WDCPDC 0.05 ,
WDCPDO 0.05 ,
WDCBBF 0.10 ,
WDCBPK 0.10 ,
WDCBML 0.10 ,
WDCBPM 0.10 ,
WDCBPE 0.10 ,
WDCBWH 0.10 ,
WDCBCN 0.10 ,
WDCBCG 0.10 ,
WDCBRI 0.10 ,
WDCBSB 0.10 ,
WDCBOS 0.10 ,
WDCBSM 0.10 ,
WDCBSO 0.10 ,
WDCBOM 0.10 ,
WDCBOO 0.10 ,
WDCBDB 0.10 ,
WDCBDC 0.10 ,
WDCBDO 0.10 ,
;FILE;
---------+---------+---------+---------+---------+---------+---------+--
OUTPUT RMARG 132 FPTFIELD 14 TABWIDTH 7 ;
CEDIT WDGOL;
CHANGE
WDCPBF 0.05 ,
WDCPPK 0.05 ,
```

```
WDCPML 0.05 ,
WDCPPM 0.05 ,
WDCPPE 0.05 ,
WDCPWH 0.05 ,
WDCPCN 0.05 ,
WDCPCG 0.05 ,
WDCPRI 0.05 ,
WDCPSB 0.05 ,
WDCPOS 0.05 ,
WDCPSM 0.05 ,
WDCPSO 0.05 ,
WDCPOM 0.05 ,
WDCPOO 0.05 ,
WDCPDB 0.05 ,
WDCPDC 0.05 ,
WDCPDO 0.05 ,
WDCBBF 0.10 ,
WDCBPK 0.10 ,
WDCBML 0.10 ,   <-- Create elasticities for 'change' program
WDCBPM 0.10 ,
WDCBPE 0.10 ,
WDCBWH 0.10 ,
WDCBCN 0.10 ,
WDCBCG 0.10 ,
WDCBRI 0.10 ,
WDCBSB 0.10 ,
WDCBOS 0.10 ,
WDCBSM 0.10 ,
WDCBSO 0.10 ,
WDCBOM 0.10 ,
WDCBOO 0.10 ,
WDCBDB 0.10 ,
WDCBDC 0.10 ,
WDCBDO 0.10 ,
;FILE;
DORANGE;
BINDVAL CONST WDGOL;
DOCORE WDPAR=MATREP(WDPAR,WDCPBF'P, 1, 1);
DOCORE WDPAR=MATREP(WDPAR,WDCPPK'P, 2, 1);
DOCORE WDPAR=MATREP(WDPAR,WDCPML'P, 3, 1);
DOCORE WDPAR=MATREP(WDPAR,WDCPPM'P, 4, 1);
DOCORE WDPAR=MATREP(WDPAR,WDCPPE'P, 5, 1);
DOCORE WDPAR=MATREP(WDPAR,WDCPWH'P, 6, 1);
DOCORE WDPAR=MATREP(WDPAR,WDCPCN'P, 7, 1);
DOCORE WDPAR=MATREP(WDPAR,WDCPCG'P, 8, 1);
DOCORE WDPAR=MATREP(WDPAR,WDCPRI'P, 9, 1);
DOCORE WDPAR=MATREP(WDPAR,WDCPSB'P, 10, 1);
DOCORE WDPAR=MATREP(WDPAR,WDCPOS'P, 11, 1);
DOCORE WDPAR=MATREP(WDPAR,WDCPSM'P, 12, 1);
DOCORE WDPAR=MATREP(WDPAR,WDCPSO'P, 13, 1);
DOCORE WDPAR=MATREP(WDPAR,WDCPOM'P, 14, 1);
DOCORE WDPAR=MATREP(WDPAR,WDCPOO'P, 15, 1);   <-- Create TROLL 'DOCORE'
DOCORE WDPAR=MATREP(WDPAR,WDCPDB'P, 16, 1);       equations to put elasticities/
DOCORE WDPAR=MATREP(WDPAR,WDCPDC'P, 17, 1);       parameters in the display
DOCORE WDPAR=MATREP(WDPAR,WDCPDO'P, 18, 1);       matrix
DOCORE WDPAR=MATREP(WDPAR,WDCBBF'P, 1, 2);
DOCORE WDPAR=MATREP(WDPAR,WDCBPK'P, 2, 2);
DOCORE WDPAR=MATREP(WDPAR,WDCBML'P, 3, 2);
DOCORE WDPAR=MATREP(WDPAR,WDCBPM'P, 4, 2);
DOCORE WDPAR=MATREP(WDPAR,WDCBPE'P, 5, 2);
DOCORE WDPAR=MATREP(WDPAR,WDCBWH'P, 6, 2);
DOCORE WDPAR=MATREP(WDPAR,WDCBCN'P, 7, 2);
DOCORE WDPAR=MATREP(WDPAR,WDCBCG'P, 8, 2);
DOCORE WDPAR=MATREP(WDPAR,WDCBRI'P, 9, 2);
DOCORE WDPAR=MATREP(WDPAR,WDCBSB'P, 10, 2);
DOCORE WDPAR=MATREP(WDPAR,WDCBOS'P, 11, 2);
DOCORE WDPAR=MATREP(WDPAR,WDCBSM'P, 12, 2);
DOCORE WDPAR=MATREP(WDPAR,WDCBSO'P, 13, 2);
DOCORE WDPAR=MATREP(WDPAR,WDCBOM'P, 14, 2);
DOCORE WDPAR=MATREP(WDPAR,WDCBOO'P, 15, 2);
DOCORE WDPAR=MATREP(WDPAR,WDCBDB'P, 16, 2);
DOCORE WDPAR=MATREP(WDPAR,WDCBDC'P, 17, 2);
DOCORE WDPAR=MATREP(WDPAR,WDCBDO'P, 18, 2);
DO WDPAR = WDPAR;
DOCORE OPRTMAT(WDPAR,WDROW'L,WDPARC'L,0,0,-28,WDPART'L,1);
---------+---------+---------+---------+---------+---------+---------+--
END OF FILE
```

II-B.1.h. CCOPY - Copy a New File for Another Country from a US File

CCOPY takes any U.S. '.TXT' file and makes a copy substituting another country code for 'US'. The program can be used to create 'template' files for other countries/regions from U.S. files.

```
10 REM CREATED BY VERNON OLEY RONINGEN - FEB. 21, 1983
20 CLEAR 2000
30 PRINT CHR$(27);CHR$(58)
40 PRINT" C  C O P Y  BY VOR":PRINT
50 PRINT"PROGRAM TO COPY US (--.TXT) FILES FOR OTHER COUNTRIES":PRINT
60 REM ----------ASK FOR COUNTRY CODE
70 INPUT"ENTER 2 DIGIT COUNTRY CODE";C$
80 IF LEN(C$)<0 OR LEN(C$)>2 THEN GOTO 70
90 PRINT:PRINT"YOU ENTERED - ";C$:PRINT
100 PRINT:W$="Y":INPUT"OKAY (Y OR N)";W$
110 IF LEFT$(W$,1)="N" THEN 30
120 REM ----------ASK FOR EQUATION FILE
130 PRINT CHR$(27);CHR$(58)
140 PRINT:INPUT"ENTER 2 DIGIT CODE FOR TYPE OF EQUATION FILE TO BE DUPLICATED";F$
150 IF F$="" THEN 360
160 I$="B:US"+F$+".TXT"
170 X$="US"
180 O$="B:"+C$+F$+".TXT"
190 OPEN"I",1,I$
200 OPEN"O",2,O$
210 IF EOF(1) THEN 290
220 LINE INPUT #1,W$
230 PRINT:PRINT W$
240 Y$=C$
250 GOSUB 310
260 PRINT W$
270 PRINT #2,W$
280 GOTO 210
290 PRINT
300 CLOSE 1:CLOSE 2 :PRINT"FILE ";O$;" IS DONE":PRINT:GOTO 120
310 REM ----------SUBROUTINE TO REPLACE X$ WITH Y$ IN W$
320 P=INSTR(W$,X$)
330 IF P=0 THEN RETURN
340 MID$(W$,P)=Y$
350 GOTO 320
360 PRINT:PRINT"DONE":END
```

Input file

```
USOF$$
*(USPD##'N/USLFI'DEF)**USOF$$##'C
*USGCAU'DEF
```

Copied output file where US is changed to EC

```
ECOF.TXT
ECOF$$
*(ECPD##'N/ECLFI'DEF)**ECOF$$##'C
*ECGCAU'DEF
```

II-B.2. Utility Programs

II-B.2.a. TDAMSC - Convert TDAM data to SUPERCALC Data

TDAMSC takes a listing of TDAM data stored on disk via a telecommunications package and creates on intermediate file 'T.XQT' which can be read into SUPERCALC with the /X command. Thus TDAM data is inserted into SUPERCALC.

```
A>10 REM CREATED BY VERNON OLEY RONINGEN - FEB. 24, 1983
20 REM THIS PROGRAM REQUIRES A TDAM OR OTHER FILE SAVED AS **.TXT ON DISK B
30 CLEAR 5000
40 REM ----------THIS LOOP SETS EXPECTED COLUMN WIDTHS OF TABLE
50 REM CHANGE THE DATA NUMBERS IF FFERENT COLUMN WIDTHS ARE EXPECTED
60 REM A MAXIMUM OF 14 COLUMNS IN TABLE IS EXPECTED
70 DIM W(14):FOR I=1 TO 14
80 READ W(I):NEXT I
90 DATA 6, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9
100 PRINT CHR$(27);CHR$(58)
110 PRINT" T D A M   TO  S U P E R C A L C   BY VOR  (TDAMSC)":PRINT
120 PRINT"PROGRAM TO PUT TDAM (OR OTHER TABLE) ON DRIVE B INTO SUPERCALC":PRINT
130 INPUT"ENTER NAME OF TDAM TABLE ON DRIVE B (WITHOUT '.TXT')";N$
140 F$="B:"+N$+".TXT":OPEN"I",1,F$
150 DIM D$(1,50)
160 REM ----------TABLE READ IN FROM INPUT FILE
170 PRINT:PRINT"TABLE ";F$;" BEING READ":PRINT
180 I1=0
190 LINE INPUT #1,W$
200 L1=LEN(W$)
210 I1=I1+1
220 X=0
230 FOR J1=1 TO 14
240 X=X+W(J1)
250 IF X>L1 THEN 270
260 NEXT J1
270 J1=J1-1
280 PRINT W$
290 D$(1,1)=W$
300 LINE INPUT #1, W$
310 PRINT W$
320 IF EOF(1) THEN 360
330 I1=I1+1
340 D$(1,I1)=W$
350 GOTO 300
360 I1=I1+1
370 D$(1,I1)=W$
380 PRINT CHR$(27);CHR$(58)
390 PRINT"SUPERCALC CREATION PROGRAM BEING CREATED":PRINT
400 CLOSE 1:O$="T.XQT":OPEN"O",1,O$
410 FOR I=1 TO I1
420 B1=1
430 FOR J=1 TO J1
440 P$=MID$(D$(1,I),B1,W(J))
450 B1=B1+W(J)
460 IF VAL(P$)=0 THEN PRINT #1,CHR$(34);P$  ELSE PRINT #1,VAL(P$)
470 IF VAL(P$)=0 THEN PRINT CHR$(34);P$ ELSE PRINT VAL(P$)
480 NEXT J
490 PRINT #1,"=A";I+1
500 PRINT "=A";I+1
510 NEXT I
520 GOSUB 590
```

```
530 V$="/S,"+CHR$(27)+"Z"+"T,A"
540 PRINT #1,V$:PRINT V$
550 PRINT #1,"/Q,Y":PRINT "/Q,Y"
560 CLOSE 2:CLOSE 1
570 SYSTEM
580 END
590 REM ----------PUT SUPERCALC STATEMENTS YOU WANT EXECUTED IN THIS SUBROUTINE
600 RETURN
10 REM CREATED BY VERNON OLEY RONINGEN - FEB. 24, 1983
20 REM THIS PROGRAM REQUIRES A SUPERCALC PRN FILE SAVE AS **.PRN ON DISK B
30 CLEAR 5000
40 REM ----------THIS LOOP SETS EXPECTED COLUMN WIDTHS OF TABLE
50 REM CHANGE THE DATA NUMBERS IF FFERENT COLUMN WIDTHS ARE EXPECTED
60 REM A MAXIMUM OF 14 COLUMNS IN TABLE IS EXPECTED
70 DIM W(14):FOR I=1 TO 14
80 READ W(I):NEXT I
90 DATA 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9
100 PRINT CHR$(27);CHR$(58)
110 PRINT" S U P E R C A L C   P R N   TO   S U P E R C A L C   BY VOR (PRNSC)":PRINT
120 PRINT"PROGRAM TO PUT SUPERCALC '.PRN' FILE ON DRIVE B BACK INTO SUPERCALC":PRINT
130 INPUT"ENTER NAME OF SUPERCALC '.PRN' FILE ON DRIVE B (WITHOUT '.PRN')";N$
140 F$="B:"+N$+".PRN":OPEN"I",1,F$
150 DIM D$(1,50)
160 REM ----------TABLE READ IN FROM INPUT FILE
170 PRINT:PRINT"TABLE ";F$;" BEING READ":PRINT
200 I1=0
210 LINE INPUT #1,W$
220 L1=LEN(W$)
230 I1=I1+1
240 X=0
250 FOR J1=1 TO 14
260 X=X+W(J1)
270 IF X>L1 THEN 290
280 NEXT J1
290 J1=J1-1
300 PRINT W$
310 D$(1,1)=W$
320 LINE INPUT #1, W$
330 PRINT W$
340 IF LEFT$(W$,3)="END" THEN 410
350 IF EOF(1) THEN 390
360 I1=I1+1
370 D$(1,I1)=W$
380 GOTO 320
390 I1=I1+1
400 D$(1,I1)=W$
410 PRINT CHR$(27);CHR$(58)
420 PRINT"SUPERCALC CREATION PROGRAM BEING CREATED":PRINT
430 CLOSE 1:O$="T.XQT":OPEN"O",1,O$
440 FOR I=1 TO I1
450 B1=1
460 FOR J=1 TO J1
470 P$=MID$(D$(1,I),B1,W(J))
480 B1=B1+W(J)
490 IF VAL(P$)=0 THEN PRINT #1,CHR$(34);P$  ELSE PRINT #1,VAL(P$)
500 IF VAL(P$)=0 THEN PRINT CHR$(34);P$ ELSE PRINT VAL(P$)
510 NEXT J
520 PRINT #1,"=A";I+1
530 PRINT "=A";I+1
540 NEXT I
550 PRINT #1,"=A1":PRINT "=A1"
560 GOSUB 630
570 V$="/ST,A"
580 PRINT #1,V$:PRINT V$
590 PRINT #1,"/QY":PRINT "/QY"
600 CLOSE 2:CLOSE 1
610 SYSTEM
620 END
630 REM ----------PUT SUPERCALC STATEMENTS YOU WANT EXECUTED IN THIS SUBROUTINE
640 RETURN
```

Intermediate T.XQT program of SUPERCALC commands →

```
 10
"   YEAR   W
"HSPRAR   W
"HCITAR   W
"HUXTAR   W
=A 2
 20
"  .....   .
"  .....   .
"  .....   .
"  .....   .
=A 3
 30
 1960
 3960
 1192
 1094
=A 4
 40
 1961
 5725
 764
 2717
=A 5
 50
 1962
 5700
 243
```

Input TDAM listing of data

```
DSNAME='[redacted].[redacted].SPEAKEZ.MYKEEP.DATA(ARWH)'
        YEAR  WHSPRAR  WHCITAR  WHUXTAR  WHSMTAR
       .....  .......  .......  .......  .......
        1960     3960     1192     1094        0
        1961     5725      764     2717        0
        1962     5700      243     1796        0
        1963     8940      504     3460        0
        1964    11260     2213     6287        0
        1965     6079     3340     5586        0
        1966     6247      175     2233      134
        1967     7320      245     2199       35
        1968     5740     1008     2494      390
        1969     7020      850     2322        0
        1970     4920      780      969        0
        1971     5680      675     1629        0
        1972     6900      370     3193      493
        1973     6560      269     1582        0
        1974     5970     1026     1784        0
        1975     8570      714     3162        0
        1976    11000      742     5900        0
        1977     5700     1600     1775        0
        1978     8100     1176     4080        0
        1979     8100     1103     4755        0
        1980     7780      428     3845        0
        1981     8100      413     3626        0
        1982    14000      737    10000        0
END OF DATA
READY
```

This CP/M SUBMIT program can be used to help get the intermediate T.XQT file into SUPERCALC and do the appropriate file management.

```
XSUB
ERA T.CAL
SC
ERA T.XQT
PIP B:$1.CAL=T.CAL
```

```
TYPE B:T.PRN

    :   A   ::    B    ::   C    ::   D    ::   E    ::
  1:      YEAR  WHSPRAR  WHCITAR  WHUXTAR  WHSMTAR
  2:     .....  .......  .......  .......  .......
  3:      1960     3960     1192     1094        0
  4:      1961     5725      764     2717        0
  5:      1962     5700      243     1796        0
  6:      1963     8940      504     3460        0
  7:      1964    11260     2213     6287        0
  8:      1965     6079     3340     5586        0
  9:      1966     6247      175     2233      134
 10:      1967     7320      245     2199       35
 11:      1968     5740     1008     2494      390
 12:      1969     7020      850     2322        0
 13:      1970     4920      780      969        0
 14:      1971     5680      675     1629        0
 15:      1972     6900      370     3193      493
 16:      1973     6560      269     1582        0
 17:      1974     5970     1026     1784        0
 18:      1975     8570      714     3162        0
 19:      1976    11000      742     5900        0
 20:      1977     5700     1600     1775        0
 21:      1978     8100     1176     4080        0
 22:      1979     8100     1103     4755        0
 23:      1980     7780      428     3845        0
 24:      1981     8100      413     3626        0
 25:      1982    14000      737    10000        0
```

Final set of TDAM data put up in a SUPERCALC file by the execution of the T.XQT program with the /X command of SUPERCALC.

II-B.2.b. SCTROLL - Convert SUPERCALC Data to TROLL Data

SCTROLL converts SUPERCALC data (a **. PRN file) to a file of data in the TROLL data input format. This latter file can be telecommunicated to the mainframe computer and can be entered into a TROLL file.

```
10 REM CREATED BY VERNON OLEY RONINGEN - FEB. 10, 1983
20 REM ----------SAMPLE INPUT FILE REQUIRED BY THIS PROGRAM:-----------------
30 REM NAME         USQDWH    USQDCN              USQDCG
40 REM
50 REM       1971     33.45     45.56     44.44     33.31
60 REM       1972     21.44     48.66     21.44     33.33
70 REM ------------------------------------------------------------------------
80 CLEAR 5000
90 PRINT CHR$(27);CHR$(58)
100 ON ERROR GOTO 1100
110 PRINT"S C T R O L L   BY VOR":PRINT
120 PRINT:PRINT"PROGRAM TO CONVERT SUPERCALC DATA TO TROLL FORMAT"
130 PRINT"THE TROLL DATA ON THE OUTPUT FILE CAN BE TRANSMITTED TO WCC":PRINT
140 PRINT:PRINT"THE DATA FILE READ MUST BE A SUPERCALC '.PRN' FILE AS FOLLOWS:":PRINT
150 PRINT"1ST LINE - NAMES OF TROLL VARIABLES OVER DATA COLUMNS"
160 PRINT"            (NO NAME MEANS DATA COLUMN IS IGNORED)"
170 PRINT"2ND LINE - A BLANK LINE"
180 PRINT"3RD LINE - DATA FOR 1ST YEAR"
190 PRINT"4TH LINE - DATA FOR 2ND YEAR"
200 PRINT"ETC."
210 PRINT:PRINT"NOTE THAT 1ST COLUMN OF EACH DATA LINE MUST CONTAIN THE YEAR"
220 PRINT"MISSING DATA WILL BE GIVEN TROLL VALUES OF 'NA'"
230 PRINT"THE INPUT FILE CAN CONTAIN UP TO 40 DATA ROWS"
240 PRINT:PRINT
250 DIM Y(40),D(40,20),O(20)
260 REM ----------ENTER THE NAME OF THE OUTPUT FILE--------------------------
270 INPUT"ENTER NAME (EXCL. '.TXT' APPENDIX) OF OUTPUT FILE FOR DRIVE 'B'";O$
280 O$="B:"+O$+".TXT"
290 OPEN"O",1,O$:PRINT
300 PRINT#1,"BEGINNING OF FILE"
310 PRINT CHR$(27);CHR$(58)
```

```
TEST.PRN     VNAME1    VNAME2

     1970       100       150
     1971       103       155
     1972       120       161
     1973       125       153
     1974       108       142
     1975       103       132
     1976       105       139
     1977       109       143
     1978       131       142
     1979       133       149
     1980       127       158
```

← SUPERCALC data (**. PRN) input file

Output file of data in the TROLL data input format ↓

```
BEGINNING OF FILE
DEDIT    VNAME1 ,1, 1970;
DATA  100  103  120  125  108  103  105  109  131  133  127; FILE;
DEDIT    VNAME2 ,1, 1970;
DATA  150  155  161  153  142  132  139  143  142  149  158: FILE;
END OF FILE
```

II-B.2.c. SCSAS - Convert SUPERCALC Data to SAS Data

SCSAS converts SUPERCALC data to the format needed for input into SAS.

```
10 REM CREATED BY VERNON OLEY RONINGEN - FEB. 10, 1983
20 REM ----------SAMPLE INPUT FILE REQUIRED BY THIS PROGRAM----------
30 REM NAME         USQDWH   USQDCN            USQDCG
40 REM
50 REM      1971     33.45    45.56   44.44    33.31
60 REM      1972     21.44    48.66   21.44    33.33
70 REM ------------------------------------------------------------------------
80 GOSUB 1250
90 CLEAR 5000:DIM O(30)
100 ON ERROR GOTO 1230
110 PRINT" S C S A S   BY VOR":PRINT
120 PRINT:PRINT"PROGRAM TO CONVERT SUPERCALC DATA TO SAS DATA":PRINT
130 PRINT"THE SAS DATA CAN BE TRANSMITTED TO WCC":PRINT
140 PRINT:PRINT"THE INPUT DATA FILES MUST BE SUPERCALC '.PRN' FILES AS FOLLOWS:":PRINT
150 PRINT"1ST LINE - NAMES OF SAS VARIABLES OVER DATA COLUMNS"
160 PRINT"            (NO NAME MEANS A COLUMN IS IGNORED)"
170 PRINT"2ND LINE - A BLANK LINE"
180 PRINT"3RD LINE - DATA FOR 1ST YEAR"
190 PRINT"4TH LINE - DATA FOR 2ND YEAR"
200 PRINT"ETC.":PRINT
210 PRINT"NOTE THAT INPUT SUPERCALC FILE CAN HAVE UP TO 30 LABELED"
220 PRINT"VARIABLES AND 50 ROWS (YEARS) OF DATA":PRINT
230 DIM D$(1,50)
240 INPUT"ENTER NAME (WITHOUT '.TXT') OF SAS OUTPUT FILE";O$
250 O$="B:"+O$+".TXT"
260 OPEN "O",1,O$:PRINT
270 PRINT #1,"BEGINNING OF FILE"
280 REM ----------BEGINNING OF LOOP TO READ IN INPUT SUPERCALC FILE----------
290 GOSUB 1250
300 PRINT"----------":PRINT:I$=""
310 PRINT"ENTER NAME OF SUPERCALC INPUT FILE (WITHOUT '.PRN' - PRESS"
320 INPUT"RETURN TO QUIT AND CLOSE OUTPUT FILE)";I$:T$=I$
330 IF I$="" THEN GOTO 1110
340 LPRINT:LPRINT:I$="B:"+I$+".PRN":LPRINT I$
350 OPEN"I",2,I$
360 LINE INPUT #2,N$
370 LPRINT N$
380 PRINT:PRINT"SUPERCALC DATA BEING CONVERTED TO SAS DATA":PRINT
390 LINE INPUT #2,X$
400 LPRINT:PRINT
410 N1=(LEN(N$)/9)-1:I=0 'FIND NUMBER OF COLUMNS IN INPUT FILE
420 REM ----------CHECK FOR DATA COLUMNS WITHOUT VARIABLE NAMES----------
430 FOR J=1 TO N1
440 IF MID$(N$,J*9+1,9)="          "THEN GOTO 470
450 I=I+1:O(I)=J
460 PRINT"COLUMN ";J,:PRINT MID$(N$,J*9+1,9)
470 NEXT J
480 PRINT
490 J1=I
500 I=0
510 REM ----------LOOP TO READ IN INPUT FILE----------
520 I=I+1
530 LINE INPUT #2,W$
540 IF EOF(2)=-1 THEN GOTO 620
550 LPRINT W$
560 X$="NA":Y$=". ":GOSUB 1200
570 X$="ERROR":Y$="    . ":GOSUB 1200
580 D$(1,I)=W$
590 PRINT W$
600 GOTO 520
```

```
610 REM ----------BEGIN TO WRITE OUT SAS DATASET------------------------------
620 LPRINT W$:X$="NA":Y$=". ":GOSUB 1200
630 X$="ERROR":Y$="    . ":GOSUB 1200
640 PRINT W$
650 D$(1,I)=W$
660 I1=I:PRINT:X$=" * DATA FROM "+I$:PRINT X$:PRINT #1,X$
670 LPRINT:LPRINT X$
680 PRINT"DATA ";T$;" ;"
690 LPRINT "DATA ";T$;" ;"
700 PRINT #1,"DATA ";T$;" ;"
710 K0=0:L$="INPUT YEAR 1-9 "
720 K5=0
730 REM ----------WRITE VARIABLES IN SAS FORMAT UP TO 7 AT A TIME--------------
740 FOR K=1 TO J1 STEP 7
750 K5=K5+1
760 J2=K
770 IF J1<K+6 THEN J3=J1 ELSE J3=K+6
780 REM ----------WRITE VARIABLE NAMES AND COLUMN LOCATIONS--------------------
790 FOR J=J2 TO J3
800 K0=K0+1:K3=K0*9+1:W$=MID$(N$,O(J)*9+1,9)+STR$(K3)+"-"+RIGHT$(STR$(K3+8),2)
810 GOSUB 1160
820 NEXT J
830 W$=" ;":GOSUB 1160
840 IF K=1 THEN GOTO 910
850 W$=""+RIGHT$(STR$(K5),1)+" ":GOSUB 1160
860 K0=0
870 FOR J=J2 TO J3
880 K0=K0+1:K3=K0*9+1:W$=MID$(N$,O(J)*9+1,9)+STR$(K3)+"-"+RIGHT$(STR$(K3+8),2)
890 GOSUB 1160
900 NEXT J
910 NEXT K
920 W$=" ":GOSUB 1160
930 PRINT L$:PRINT #1,L$:LPRINT L$
940 LPRINT "LIST ; CARDS ;"
950 PRINT"LIST ; CARDS ;":PRINT #1,"LIST ; CARDS ;"
960 FOR K=1 TO J1 STEP 7
970 J2=K
980 IF J1<K+6 THEN J3=J1 ELSE J3=K+6
990 FOR I=1 TO I1
1000 X$=D$(1,I)
1010 W$=MID$(X$,1,9)
1020 FOR J=J2 TO J3
1030 W$=W$+MID$(X$,O(J)*9+1,9)
1040 NEXT J
1050 PRINT W$
1060 LPRINT W$
1070 PRINT #1,W$
1080 NEXT I
1090 NEXT K
1100 CLOSE 2:GOTO 290
1110 REM
1120 PRINT #1,"END OF FILE":LPRINT
1130 CLOSE 2:CLOSE 1
1140 PRINT:PRINT"FILE ";O$;" COMPLETED":PRINT:END
1150 REM ----------SUBROUTINE TO PRINT SAS DATA LINE IF NO. CHARS. > 72 --------
1160 IF LEN(L$)+LEN(W$)<=72 THEN GOTO 1180
1170 PRINT L$:PRINT #1,L$:LPRINT L$:L$=""
1180 L$=L$+W$:RETURN
1190 REM ----------SUBROUTINE TO REPLACE STRING X$ WITH STRING Y$ IN STRING W$ -
1200 P0=INSTR(W$,X$)
1210 IF P0=0 THEN RETURN
1220 MID$(W$,P0)=Y$:GOTO 1200
1230 PRINT:PRINT"FILE ";I$;" NOT FOUND":PRINT:CLOSE 1:END
1240 REM ----------SUBROUTINE TO CLEAR SCREEN----------------------------------
1250 PRINT CHR$(27);CHR$(58):RETURN
```

SAS output data file created from
input data file shown with SCTROLL program

```
BEGINNING OF FILE
 * DATA FROM B:TEST.PRN
DATA TEST ;
INPUT YEAR 1-9    VNAME1 10-18   VNAME2 19-27 ;
LIST ; CARDS ;
     1970       100       150
     1971       103       155
     1972       120       161
     1973       125       153
     1974       108       142
     1975       103       132
     1976       105       139
     1977       109       143
     1978       131       142
     1979       133       149
     1980       127       158
END OF FILE
```

| NOTATION FOR ERS GRAIN, OILSEED, AND LIVESTOCK (GOL) MODEL : SYMBOL AND VARIABLE NAMES CONTAIN UP TO 8 CHARACTERS AND ARE FOLLOWED BY A SUFFIX WHICH SHOWS THE DECLARATION (E.G. CONSTANT, EXOGENOUS VARIABLE, ETC.). THE FIRST 2 CHARACTERS ARE THE COUNTRY CODE AND THE NEXT 2, AN EQUATION 'TYPE' CODE. THE NEXT 2 CHARACTERS ARE USUALLY A 2 DIGIT COMMODITY CODE. AN ELASTICITY WILL HAVE 2 MORE CHARACTERS INDICATING THE CODE TO WHICH THE ELASTICITY RELATES. GENERALLY, THE NUMBER OF CHARACTERS IN A SYMBOL HAS A MEANING: 5 CHAR. = COUNTRY SPECIFIC VARIABLE, 6 CHAR. = COUNTRY AND COMMODITY SPECIFIC VARIABLE, 7 CHAR.(ENDING WITH 'I') = EQUATION INTERCEPT, 8 CHAR. = COEFFICIENT/ELASTICITY. COMMODITY CODES ARE: BF = BEEF+VEAL, PK = PORK, ML = MUTTON+LAMB(+GOAT), DM = DAIRY-MILK, PM = POULTRY-MEAT, PE = POULTRY-EGGS, WH= WHEAT, CN = CORN, CG = OTHER COARSE GRAINS, RI = RICE, SB= SOYBEANS, OS = OTHER OILSEEDS, SM = SOYMEAL, SC = SOYOIL, OM = OTHER MEALS, OO = OTHER OILS, DB = DAIRY-BUTTER, DC = DAIRY-CHEESE, DO = DAIRY-OTHER PRODUCTS. EQUATION 'TYPE' CODES ARE: MD = MARGIN-DOMESTIC, MT = MARGIN-TRADE, PS = PRICE-SUPPLY, AR = AREA, YD = YIELD, QS = QUANTITY-SUPPLIED, QC = QUANTITY-CRUSHED, FC = FEED COST, LN = LIVESTOCK-NUMBERS, LA = LIVESTOCK-ADDITIONS, LS = LIVESTOCK-SLAUGHTER, QF = QUANTITY-FED, QD = QUANTITY-FOOD AND OTHER DEMAND, SK = ENDING STOCKS, QT = QUANTITY-TRADED, PE = PRICE ESTIMATE (DEMAND) WITH TRADE RESTRICTIONS, PD = PRICE-DEMAND. POLICY VARIABLE CODES ARE: EQ = EXPORT QUOTA, MQ = IMPORT QUOTA, TE = TAX-EXPORTS, TM = TAX-IMPORTS, TP = TAX-PRODUCTION, TC = TAX-CONSUMPTION. |

II-B.2.d. VORPLOT - Plot SUPERCALC Data

VORPLOT is a user friendly, self prompting program which allows the user to produce plots of time series data that are stored on SUPERCALC **. PRN files.

```
10 REM CREATED BY VERNON OLEY RONINGEN - FEB. 20, 1983
20 CLEAR 9000:DIM G$(0,120)
30 P$="ABCDEFGHIJ"+CHR$(172)+CHR$(167) '----------SET PLOT CHARACTERS
40 DIM S(40,10),B$(0,10),O(30):A$=STRING$(9," ")
50 DEFINT I-L,N,R:R1=0:R2=0:R3=0
60 L8=80:L7=24 '----------SET SCREEN PARAMETERS
70 LPRINT CHR$(27);"2":GOSUB 2890 '-----PUT MENU ON SCREEN
80 PRINT" V O R P L O T   BY VOR - PROGRAM TO PLOT ANNUAL TIME SERIES DATA":PRINT
90 PRINT" 1 INPUT DATA FROM KEYBOARD";:I=1:GOSUB 260
100 PRINT" 2 INPUT DATA FROM SUPERCALC FILE";:I=2:GOSUB 260
110 PRINT" 3 INPUT DATA FROM 'DIF' FILE";:I=3:GOSUB 260
120 PRINT" 4 SELECT VARIABLES FOR PLOTTING";:I=4:GOSUB 260
130 PRINT" 5 CREATE GRAPH LAYOUT";:I=5:GOSUB 260
140 PRINT" 6 PLOT LINE GRAPH";:I=6:GOSUB 260
150 PRINT" 7 PLOT BAR GRAPH";:I=7:GOSUB 260
160 PRINT" 8 VIEW GRAPH ON SCREEN";:I=8:GOSUB 260
170 PRINT" 9 SEND GRAPH TO PRINTER";:I=9:GOSUB 260
180 PRINT"10 END":PRINT
190 INPUT"SELECT NUMBER";R
200 IF R<1 OR R>10 THEN 70
210 IF (R>3 AND R<10) AND R1=0 THEN 70
220 IF (R>4 AND R<10) AND R2=0 THEN 70
230 IF (R>5 AND R<10) AND R3=0 THEN 70
240 ON R GOSUB 280  ,590  ,930  ,1490 ,1630 ,960  ,1160 ,1280 ,2590 ,2880
250 GOTO 70
260 PRINT TAB(37)MID$(A$,I,1):RETURN
270 MID$(A$,I,1)="*":RETURN
280 R1=1:I=1:GOSUB 270  '-----SUBROUTINE TO INPUT DATA FROM KEYBOARD
290 GOSUB 2890:PRINT"ENTER DATA FROM KEYBOARD (UP TO 10 VARS.)":PRINT
300 MID$(A$,2,8)="        ":R2=0:R3=0
310 INPUT"NUMBER OF VARIABLES";J9
320 IF J9<1 OR J9>10 THEN 310
330 PRINT:INPUT"ENTER BEGINNING YEAR";X1
340 INPUT"ENTER ENDING YEAR";X9
350 I9=X9-X1+1
360 IF I9>40 OR X1>X9 THEN 330
370 PRINT:W$="Y":PRINT"NO. OF VARS. = ";J9
380 PRINT"BEGINNING YEAR = ";X1
390 PRINT"ENDING YEAR = ";X9
400 PRINT:INPUT"OKAY (Y OR N)";W$
410 IF LEFT$(W$,1)="N" THEN 290
420 FOR I=1 TO I9:S(I,0)=X1+I-1:NEXT I
430 FOR J=1 TO J9
440 GOSUB 2890: PRINT"ENTER NAME (9 CHAR. OR LESS) FOR"
450 PRINT"VARIABLE NO. ";J;:INPUT B$(0,J):PRINT
460 FOR I=1 TO I9
470 PRINT"VALUE FOR OBS. NO. ";I;:INPUT S(I,J)
480 NEXT I
490 GOSUB 2890:PRINT B$(0,J):PRINT
500 FOR I=1 TO I9:PRINT I;S(I,J),:NEXT I:PRINT
510 W$="N":INPUT"CORRECTIONS (Y OR N)";W$
520 IF LEFT$(W$,1)="Y" THEN 530 ELSE 570
530 PRINT:INPUT"WHICH OBS.";I1
540 IF I1<1 OR I1>I9 THEN 530
550 PRINT:INPUT"CORRECT VALUE";S(I1,J)
560 GOTO 490
570 NEXT J
580 RETURN
590 R1=1:I=2:GOSUB 270  '-----SUBROUTINE TO INPUT DATA FROM SUPERCALC FILE
600 MID$(A$,3,7)="       ":R2=0:R3=0
610 MID$(A$,1,1)=" "
620 GOSUB 2890:PRINT"ENTER DATA FROM SUPERCALC FILE"
630 PRINT"UP TO 10 LABELED VARIABLES CAN BE ENTERED FROM A FILE":PRINT
```

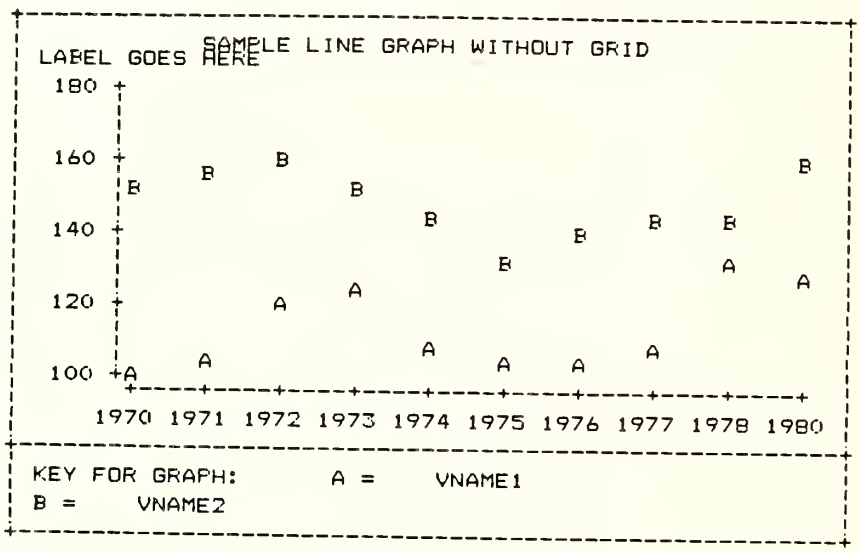
```
640 INPUT"ENTER NAME OF FILE (WITHOUT '.PRN')";F$:F$="B:"+F$+".PRN"
650 OPEN"I",1,F$:FOR I=1 TO 11:O(I)=0:NEXT I
660 PRINT:PRINT"READING FILE":PRINT
670 J9=0:LINE INPUT#1,W$:N=LEN(W$)/9
680 FOR J=2 TO N
690 IF MID$(W$,(J-1)*9+1,9)="          "THEN 720
700 J9=J9+1:B$(0,J9)=MID$(W$,(J-1)*9+1,9):O(J)=1
710 PRINT J9,B$(0,J9)
720 NEXT J
730 LINE INPUT#1,W$
740 I9=0:PRINT
750 I8=0
760 LINE INPUT#1,W$:I9=I9+1
770 IF EOF(1)THEN CLOSE 1:I9=I9-1:I8=1
780 S(I9,0)=VAL(MID$(W$,1,9)):J=0
790 PRINT S(I9,0),
800 FOR I=2 TO N STEP 1:IF O(I)=0 THEN 860
810 J=J+1
820 U$=MID$(W$,(I-1)*9+1,9)
830 GOSUB 880
840 S(I9,J)=VAL(U$)
850 PRINT J;") ";S(I9,J),
860 NEXT I:PRINT
870 IF I8=0 THEN GOTO 760 ELSE RETURN
880 Z$=U$
890 IF INSTR(Z$,"-")=0 THEN RETURN
900 IF LEFT$(Z$,1)=" "THEN Z$=RIGHT$(Z$,LEN(Z$)-1)ELSE 920
910 GOTO 900
920 U$=Z$:RETURN
930 R1=1:I=3:GOSUB 270  'INPUT DATA FROM DIF FILE
940 RETURN
950 REM ----------SUBROUTINE TO PUT LINE DATA ON GRAPH
960 GOSUB 2890:PRINT"PUT LINE DATA ON GRAPH":I=6:GOSUB 270  'PUT LINE DATA ON GRAPH
970 W$="N":PRINT:INPUT"CONNECT POINTS (Y OR N)";W$
980 FOR J=1 TO J9
990 IF O(J)=0 THEN 1140
1000 FOR I=1 TO I9
1010 K4=K3+4+(I-1)*5+1
1020 M6=S(I,J)
1030 N=N2-(INT(((M6-M1+M5/10)/(M9-M1+M5/5))*N1+5))
1040 MID$(G$(0,N),K4,1)=MID$(P$,J,1)
1050 IF LEFT$(W$,1)="Y" THEN 1060 ELSE 1130
1060 IF I=I9 THEN 1130
1070 M7=(S(I+1,J)-S(I,J))/5
1080 FOR I1=1 TO 4
1090 M8=M6+I1*M7
1100 N=N2-(INT(((M8-M1+M5/10)/(M9-M1+M5/5))*N1+5))
1110 IF M8>=(M9-(N-5)*M5/5) THEN MID$(G$(0,N),K4+I1,1)=MID$(P$,11,1) ELSE MID$(G$(0,
1120 NEXT I1
1130 NEXT I                                                    N),K4+I1,1)=MID$(P$,12,1)
1140 NEXT J:RETURN
1150 REM ----------SUBROUTINE TO PUT BARS ON GRAPH
1160 GOSUB 2890:PRINT"PLOTTING BARS ON GRAPH":J7=0:I=7:GOSUB 270  'PUT BARS ON GRAPH
1170 FOR J7=1 TO J9
1180 IF O(J7)=0 THEN 1260
1190 FOR I=1 TO I9
1200 K5=K3+4+(I-1)*5+1
1210 N=N2-(INT(((S(I,J7)-M1+M5/10)/(M9-M1+M5/5))*N1+5))
1220 FOR J=N TO N1+5
1230 W$=MID$(P$,J7,1)
1240 MID$(G$(0,J),K3+4+(I-1)*5+1,3)=W$
1250 NEXT J,I
1260 NEXT J7:RETURN
1270 REM ----------SUBROUTINE TO VIEW PART OF GRAPH ON SCREEN
1280 GOSUB 2890:PRINT"PRINT PART OF GRAPH ON SCREEN":PRINT:I=8:GOSUB 270  'PRINT ON SCREEN
1290 PRINT"MAXIMUM = ";M9:PRINT"MINIMUM = ";M1
1300 PRINT"STEP INTERVAL = ";M5
1310 PRINT:PRINT"BEGINNING YEAR = ";S(1,0)
1320 PRINT "ENDING YEAR = ";S(I9,0)
1330 PRINT:PRINT"SELECT YEAR FOR LEFT OF GRAPH"
1340 INPUT"YEAR";Z1
```

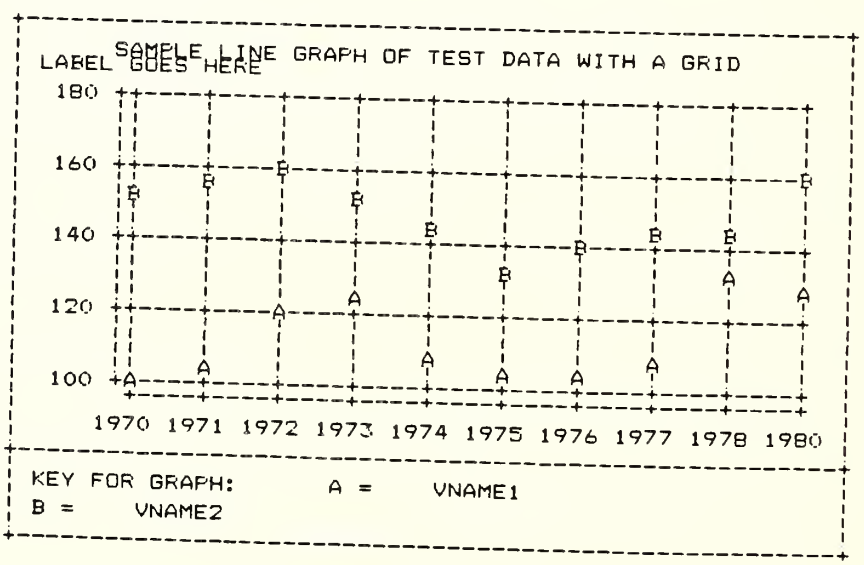
```
1350 IF Z1<S(1,0) OR Z1>S(I9,0) THEN 1340
1360 J1=Z1
1370 M3=M9
1380 IF M3>M9 OR M3<M1 THEN 1370
1390 N5=INT((M9-M3+M5/10)/(M5/5))+5
1400 FOR J=1 TO I9
1410 IF J1=S(J,0) THEN 1430
1420 NEXT J:I1=1:GOTO 1440
1430 I1=J
1440 GOSUB 2890:FOR N=N5 TO N5+L7-2
1450 PRINT MID$(G$(0,N),4,K3+1);MID$(G$(0,N),K3+2+(I1-1)*5+3,L8-K3-2)
1460 NEXT N
1470 PRINT MID$(G$(0,N1+7),K3+1+(I1-1)*5,L8-K3);:INPUT W$:RETURN
1480 REM ----------SUBROUTINE TO SELECT VARIABLES FOR PLOTTING
1490 GOSUB 2890:R2=1:I=4:GOSUB 270  :PRINT"SELECT VARIABLES FOR PLOTTING":PRINT
1500 R3=0:FOR I=5 TO 9:MID$(A$,I,1)=" ":NEXT I
1510 FOR J=1 TO J9:O(J)=0:NEXT J
1520 PRINT"BEGINNING YEAR = ";S(1,0),"ENDING YEAR = ";S(I9,0):PRINT
1530 FOR J=1 TO J9:PRINT J;") ";B$(0,J),:NEXT J:PRINT:PRINT:K=0
1540 INPUT"SELECT VARIABLE FOR PLOT (ENTER '0' WHEN DONE)";L1:K=K+1
1550 IF L1=0 THEN 1580
1560 IF L1<0 OR L1>J9 THEN 1540
1570 O(L1)=1:GOTO 1540
1580 PRINT:PRINT"VARIABLES SELECTED ARE:",:FOR J=1 TO 10
1590 IF O(J)=0 THEN 1610
1600 PRINT B$(0,J),
1610 NEXT J:PRINT:W$="Y":PRINT
1620 INPUT"OK (Y OR N)";W$:IF LEFT$(W$,1)="N" THEN 1490 ELSE RETURN
1630 GOSUB 2100 :GOSUB 2430 :GOSUB 2480 '-----SUBROUTINE TO CREATE GRAPH BACKGROUND
1640 GOSUB 2890:PRINT"CREATING GRAPH LAYOUT"
1650 R3=1:I=5:GOSUB 270
1660 K3=LEN(STR$(M9))
1670 K0=(I9-1)*5+1
1680 K1=2+K3+2+K0+3
1690 N2=5+N1+5
1700 W$=STRING$(K1," "):MID$(W$,1)="!":MID$(W$,K1,1)="!"
1710 FOR N=1 TO N2
1720 GOSUB 2080 :NEXT N
1730 W$=STRING$(K1,"-"):MID$(W$,1)="+":MID$(W$,K1,1)="+"
1740 N=1:GOSUB 2080 :N=N2:GOSUB 2080
1750 N=3:GOSUB 2090
1760 MID$(W$,(K1-LEN(T$))/2)=T$:GOSUB 2080
1770 N=4:GOSUB 2090
1780 MID$(W$,3)=Y$:GOSUB 2080
1790 GOSUB 2020 :GOSUB 2050
1800 FOR N=6 TO N1+5:GOSUB 2090
1810 Y=M9-(N-6)*M5/5
1820 IF ((Y/M5)-INT(Y/M5))=0 THEN 1830 ELSE 1890
1830 Z=Y:GOSUB 2570 :Y=Z
1840 MID$(W$,3+K3-LEN(STR$(Y)))=STR$(Y)
1850 MID$(W$,K3+4,1)="+"
1860 IF J5=0 THEN 1900
1870 MID$(W$,K3+5)=Y$
1880 GOTO 1920
1890 MID$(W$,K3+4,1)="!"
1900 IF I5=0 THEN 1920
1910 MID$(W$,K3+5)=T$
1920 GOSUB 2080
1930 NEXT N
1940 N=N1+6:GOSUB 2090
1950 MID$(W$,K3+5)=Y$:GOSUB 2080
1960 Y$=STRING$(K0+4," ")
1970 FOR I=1 TO I9
1980 MID$(Y$,(I-1)*5+1)=STR$(INT(S(I,0))):NEXT I
1990 N=N1+8:GOSUB 2090
2000 MID$(W$,K3+2)=Y$:GOSUB 2080
2010 RETURN
2020 Y$=STRING$(K0,"-") 'CREATE HORIZONTAL GRAPH LINES
2030 FOR K=1 TO K0 STEP 5
2040 MID$(Y$,K,1)="+":NEXT K:RETURN
2050 T$=STRING$(K0," ") 'CREATE VERTICAL GRAPH LINES
2060 FOR K=1 TO K0 STEP 5
2070 MID$(T$,K,1)="!":NEXT K:RETURN
2080 G$(0,N)=W$:RETURN
2090 W$=G$(0,N):RETURN
```

```
2100 GOSUB 2890:PRINT"CALCULATING MAX-MIN VALUES FOR PLOT DATA"
2110 V1=9E+37:V9=-9E+37
2120 FOR I=1 TO I9:FOR J=1 TO J9              'SELECT SCALE VALUES FOR GRAPH
2130 IF O(J)=0 THEN 2160
2140 IF S(I,J)>V9 THEN V9=S(I,J)
2150 IF S(I,J)<V1 THEN V1=S(I,J)
2160 NEXT J,I
2170 GOSUB 2890:PRINT"MAXIMUM = ";V9
2180 PRINT"MINIMUM = ";V1
2190 PRINT"DIFFERENCE = ";V9-V1:PRINT
2200 INPUT"SELECT TOP OF Y SCALE ";M9
2210 IF M9<V9 THEN 2200
2220 INPUT"SELECT BOTTOM OF Y SCALE ";M1
2230 IF M1>V1 THEN 2220
2240 IF M1>=M9 THEN 2170
2250 INPUT"SELECT STEP INTERVAL FOR Y SCALE ";M5
2260 IF M5=0 OR M5<((M9-M1)/100) THEN 2250
2270 IF ((M9/M5)-INT(M9/M5))<>0 THEN 2250
2280 IF ((M1/M5)-INT(M1/M5))<>0 THEN 2250
2290 IF (M9-M1)<M5 THEN 2250
2300 Z=M9:GOSUB 2570 :M9=Z
2310 Z=M1:GOSUB 2570 :M1=Z
2320 Z=M5:GOSUB 2570 :M5=Z
2330 N1=((M9-M1)/M5)*5+1
2340 IF N1>100 THEN 2250
2350 PRINT:PRINT"TOP OF Y SCALE = ";M9
2360 PRINT"MAXIMUM DATA VALUE = ";V9
2370 PRINT"MINIMUM DATA VALUE = ";V1
2380 PRINT"BOTTOM OF Y SCALE = ";M1
2390 PRINT"STEP INTERVAL FOR Y SCALE = ";M5
2400 PRINT"NUMBER OF GRAPH LINES = ";N1
2410 PRINT:W$="Y":INPUT"OK (Y OR N)";W$
2420 IF LEFT$(W$,1)="N" THEN 2170 ELSE RETURN
2430 GOSUB 2890:PRINT"ENTER TITLE FOR GRAPH":T$="":Y$="":INPUT T$:PRINT
2440 INPUT"ENTER LABEL FOR Y SCALE";Y$:PRINT
2450 PRINT"TITLE IS:":PRINT T$:PRINT:PRINT"LABEL IS:  ";Y$:PRINT
2460 W$="Y":INPUT"OK (Y OR N)";W$
2470 IF LEFT$(W$,1)="N" THEN 2430 ELSE RETURN
2480 GOSUB 2890:W$="N":INPUT"HORIZONTAL (X) GRID (Y OR N) ";W$ 'SELECT GRID FOR GRAPH
2490 IF LEFT$(W$,1)="Y" THEN J5=1 ELSE J5=0:W$="N"
2500 INPUT"VERTICAL (Y) GRID (Y OR N) ";W$
2510 IF LEFT$(W$,1)="Y" THEN I5=1 ELSE I5=0
2520 PRINT:PRINT"YOU HAVE CHOSEN:"
2530 PRINT"HORIZONTAL GRID ";:IF J5=1 THEN PRINT "Y" ELSE PRINT "N"
2540 PRINT"VERTICAL GRID ";:IF I5=1 THEN PRINT "Y" ELSE PRINT "N"
2550 W$="Y":PRINT:INPUT"OK (Y OR N)";W$
2560 IF W$="N" THEN 2480 ELSE RETURN
2570 Z=INT(Z*100)/100:RETURN
2580 REM ----------SUBROUTINE TO SEND GRAPH TO PRINTER
2590 GOSUB 2890:PRINT"GRAPH GOING TO PRINTER":I=9:GOSUB 270  'PRINT GRAPH
2600 LPRINT CHR$(27);"1"
2610 N4=8 'NO. OF SPACES BEFORE AND AFTER GRAPH
2620 GOSUB 2820
2630 FOR N=1 TO N2
2640 LPRINT MID$(G$(0,N),1,132)
2650 NEXT N:GOSUB 2710
2660 IF K1<=132 THEN RETURN
2670 FOR N=1 TO N2
2680 LPRINT MID$(G$(0,N),133,K1)
2690 NEXT N
2700 GOSUB 2820 :RETURN
2710 IF K1>132 THEN K3=132 ELSE K3=K1
2720 GOSUB 2830 :LPRINT W$:T$="KEY FOR GRAPH:       "
2730 FOR J=1 TO J9
2740 IF O(J)=0 THEN 2780
2750 Y$=MID$(P$,J,1)+" = "+B$(0,J)+"        "
2760 IF (LEN(T$)+LEN(Y$)+4)>K3 THEN GOSUB 2850
2770 T$=T$+Y$
2780 NEXT J
2790 GOSUB 2850
2800 GOSUB 2840 :LPRINT W$
2810 GOSUB 2820 :RETURN
```

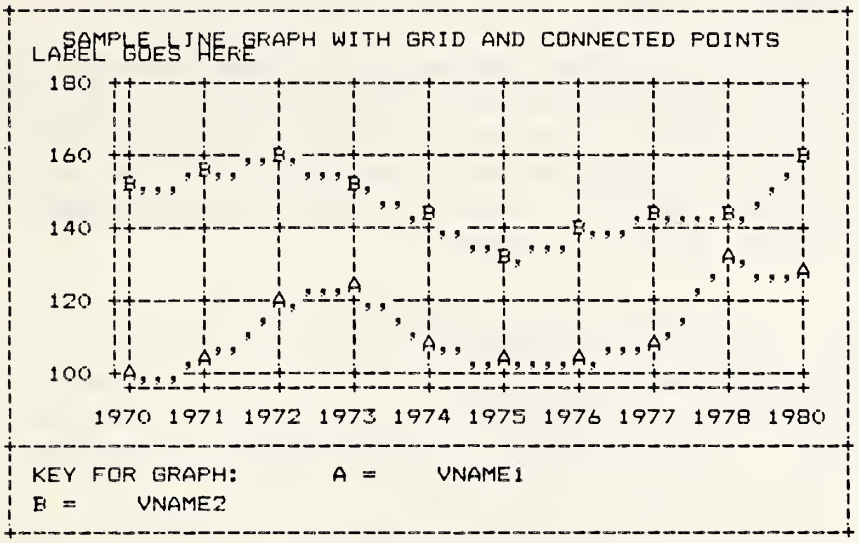
```
2820 FOR I=1 TO N4:LPRINT:NEXT I:RETURN
2830 W$=STRING$(K3," "):MID$(W$,1)="!":MID$(W$,K3,1)="!":RETURN
2840 W$=STRING$(K3,"-"):MID$(W$,1)="+":MID$(W$,K3,1)="+":RETURN
2850 GOSUB 2830                                    '-----SUBROUTINE TO CREATE BORDER FOR GR
2860 MID$(W$,3)=T$                                 '-----SUBROUTINE TO CREATE LINE FOR GRAP
2870 LPRINT W$:T$="":GOSUB 2830:LPRINT W$:RETURN
2880 END
2890 PRINT CHR$(27);CHR$(58):RETURN '-----SUBROUTINE TO CLEAR SCREEN
```

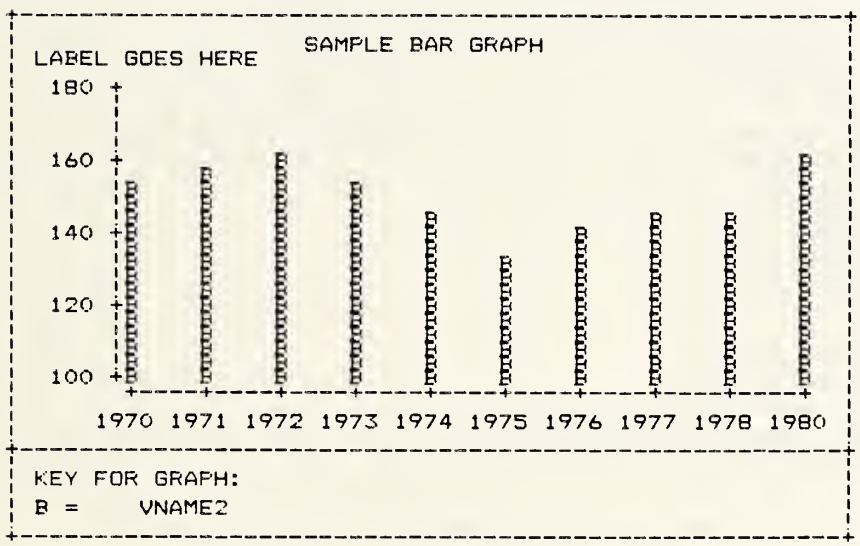
The format of the input file is that of the input file to the SCTROLL program.





Sample VORPLOT plots of the data in the input data file shown with the SCTROLL program.





The program offers several options including grids, bar plots, and marks between time series data points. The program allows the user to view part of the plot on the CRT before sending it to the printer.

II-B.2.e. SCMERGE - Merge Two SUPERCALC Files

SCMERGE allows the user to take two SUPERCALC files with the same row-colum structure but different data and merge them to form a new SUPERCALC file. A user inserted arithmetic routine sets the data merging conditions. This program is useful in those circumstances where two SUPERCALC files and some simple arithmetic imply a third file. The output file is labeled SCM.XQT and is read into SUPERCALC by the /X command.

```
10 REM PROGRAM BY VERNON OLEY RONINGEN, MAR. 15, 1983
20 CLEAR 5000
30 PRINT CHR$(27);CHR$(58)
40 PRINT" S C M E R G E   BY VOR":PRINT
50 PRINT"PROGRAM TO MERGE 2 SUPERCALC FILES WITH SIMPLE ARITHMETIC"
60 PRINT"A 3RD FILE IS PRINTED WHICH IS THE SUM, DIFF., ETC. OF THE "
70 PRINT"2 FILES AND WHICH CAN BE LOADED INTO SUPERCALC BY THE /X COMMAND"
80 PRINT"THE 3RD FILE IS NAMED 'SCM'":PRINT
90 PRINT"THE FILES TO BE READ FROM DRIVE B IN MUST BE SUPERCALC 'CONTENT' FILES":PRINT
100 INPUT"ENTER NAME OF 1ST SC CONTENT FILE";F$
110 INPUT"ENTER NAME OF 2ND SC CONTENT FILE":G$
120 P$="B:"+F$+".PRN":OPEN"I",1,P$
130 Q$="B:"+G$+".PRN":OPEN"I",2,Q$
140 OPEN"O",3,"B:SCM.XQT"
150 A$=""
160 IF EOF(1) THEN 520
170 IF EOF(2) THEN 520
180 PRINT
190 LINE INPUT #1,W$
200 LINE INPUT #2,X$
210 P0=INSTR(W$,"=")
220 IF P0=0 THEN 490
230 P0=INSTR(W$," ")-1
240 P1=INSTR(W$,"=")+2
250 P2=INSTR(X$,"=")+2
260 P3=LEN(W$)
270 P4=LEN(X$)
280 S$=RIGHT$(W$,P3-P1+1)
290 T$=RIGHT$(X$,P4-P2+1)
300 M$="="+LEFT$(W$,P0)
310 IF VAL(LEFT$(S$,1))=0 AND LEFT$(S$,1)<>"0" THEN 330 ELSE 320
320 IF LEFT$(S$,1)=CHR$(34) THEN 330 ELSE 350
330 N$=S$
340 GOTO 420
350 REM ----------SUBROUTINE TO ADD, SUBTRACT, ETC. TWO VALUES
360 Z1=VAL(S$)
370 Z2=VAL(T$)
380 REM -----THIS IS ARITHMETIC STATEMENT TO CHANGE FOR DIFFERENT OPERATIONS
390 Z=Z1-Z2
400 REM------------------------------------------------------
410 N$=STR$(Z)
420 PRINT W$;
430 PRINT TAB(40)M$
440 PRINT X$;
450 PRINT TAB(40)N$
460 IF A$=RIGHT$(M$,1) THEN 470 ELSE PRINT #3,M$
470 PRINT #3,N$
480 A$=RIGHT$(M$,1)
490 GOTO 160
500 GOTO 520
510 PRINT:PRINT"FILES ";F$;" AND ";G$;" ARE NOT IN THE SAME FORMAT":PRINT
520 PRINT #3,"=A1"
530 PRINT:PRINT"FILE B:SCM.XQT IS READY"
540 CLOSE 1:CLOSE 2:CLOSE 3:END
```

II-B.2.f. SCTHIN - Thin (Select Out Data) SUPERCALC File

SCTHIN is a simple program which allows a SUPERCALC sub-file to be created from a master file by a user supplied selection criterion. This program can, for example, create a file which contains only numbers greater than 1 from a file which contains a whole range of numbers. The output file is labeled T.XQT and can be read into SUPERCALC by the /X command.

```
10 REM PROGRAM BY VERNON OLEY RONINGEN, MAR. 15, 1983
20 CLEAR 5000
30 PRINT CHR$(27);CHR$(58)
40 PRINT" S C T H I N  BY VOR":PRINT
50 PRINT"PROGRAM TO THIN A SUPERCALC FILE BY A SIMPLE CRITERION
60 PRINT"A THINNED FILE IS PRINTED ON DISK"
70 PRINT"WHICH CAN BE LOADED INTO SUPERCALC BY THE /X COMMAND"
80 PRINT"THE FILE IS NAMED 'T'":PRINT
90 PRINT"THE FILE TO BE READ FROM DRIVE B IN MUST BE SUPERCALC 'CONTENT' FILE":PRINT
100 INPUT"ENTER NAME OF SC CONTENT FILE";F$
110 P$="B:"+F$+".PRN":OPEN"I",1,P$
120 OPEN"O",3,"B:T.XQT"
130 A$=""
140 IF EOF(1) THEN 400
150 PRINT
160 LINE INPUT #1,W$
170 P0=INSTR(W$,"=")
180 IF P0=0 THEN 390
190 P0=INSTR(W$," ")-1
200 P1=INSTR(W$,"=")+2
210 P3=LEN(W$)
220 S$=RIGHT$(W$,P3-P1+1)
230 M$="="+LEFT$(W$,P0)
240 IF VAL(LEFT$(S$,1))=0 AND LEFT$(S$,1)<>"0" AND LEFT$(S$,1)<>"." THEN 260 ELSE 250
250 IF LEFT$(S$,1)=CHR$(34) THEN 260 ELSE 280
260 N$=S$
270 GOTO 340
280 REM ----------ROUTINE TO SELECT VALUE FOR NEW SC FILE
290 Z=VAL(S$)
300 REM -----THIS IS ARITHMETIC STATEMENT TO CHANGE FOR DIFFERENT OPERATIONS
310 REM ---THIS IS SELECTION CRITERION STATEMENT
320 IF (Z < 1) THEN N$=CHR$(34)+"-" ELSE N$=STR$((INT(((Z)+.05)*10))/10)
330 REM------------------------------------------------------
340 PRINT W$
350 PRINT M$;"  ";N$
360 IF A$=RIGHT$(M$,1) THEN 370 ELSE PRINT #3,M$
370 PRINT #3,N$
380 A$=RIGHT$(M$,1)
390 GOTO 140
400 PRINT #3,"=A1"
410 PRINT:PRINT"FILE B:T.XQT IS READY"
420 CLOSE 1:CLOSE 2:CLOSE 3:END
```

II-C. Templates and Input Matrices for the Creation of GOL Equations on the Micro Computer.

The following pages contain 'template' equations and input matrices that can be used to create all of the equations groups for a standard model (USGOL), a reduced form country/region model (RWGOL), and the world market clearing mechani (WDGOL). These inputs must be used with the programs documented above to create GOL statements and equations. In some cases, the full GOL set of statements is given.

The following listing (GOLSUM) shows the equation groups, the input files needed for the equation group, the program needed to create the equations/statem for the equation group, and the output files created from the input files. The pages following this summary contain listings of the input files for each equati group.

Examples of the use of the input files to create the output files for EQWRITE an EQDUPLIC are contained in the documentation for those programs (previously liste Files requiring WORDSTAR (trademark) are final output files which must be created by a word processing program.

The programs require an input ****.TXT file and an input matrix file (created in SUPERCALC) labelled ****. PRN. The SUPERCALC creation files (marked by ****. CAL) are also listed since they most likely will be used to change the ****. PRN files. In some cases, the ****. CAL files contain formulas to impose, for example, symetry conditions on the elasticities in the **. PRN files.

The input template files can be modified to different equations structures via a word processing program. The input matrix files can be modified to different equation structures and elasticity values via SUPERCALC editing. The input file listings below serve as a guide to correct syntax required by files input to the EQWRITE and EQDUPLIC programs under a wide variety of equation forms.

GOLSUM SUMMARY OF GOL TROLL STATEMENT CREATION PROGRAMS

		---PROGRAM TO BE USED----			
EQUATION TYPE	INPUT DATA FILE	EQWRITE	EQDUPLIC	WORDSTAR	OUTPUT FILE
STANDARD COUNTRY MODEL (USGOL)					
DOMESTIC MARGIN	USMD.TXT	1			USMDR.TXT
	USMD.CAL				USMDC.TXT
	USMD.PRN	1			
TRADE MARGIN	USMT.TXT	1			USMTR.TXT
	USMT.CAL				USMTC.TXT
	USMT.PRN	1			
SUPPLY PRICE	USPS.TXT		1		USPSR.TXT
	USPS.CAL				
	USPS.PRN		1		
CROP AREA	USTTRL.TXT			1	USTTRL.TXT
	USARTT.TXT			1	USARTT.TXT
	USAR.TXT	1			USARR.TXT
	USAR.CAL				USARC.TXT
	USAR.PRN	1			

EQUATION TYPE	INPUT DATA FILE	---PROGRAM TO BE USED---- EQWRITE	EQDUPLIC	WORDSTAR	OUTPUT FILE
CROP YIELD	USYD.TXT	1			USYDR.TXT
	USYD.CAL				USYDC.TXT
	USYD.PRN	1			
CROP SUPPLY	USQS.TXT		1		USQSR.TXT
	USQS.CAL				
	USQS.PRN		1		
OILSEED PRODUCTS	USOILS.TXT			1	USOILS.TXT
FEED COSTS	USFC.TXT		1		USFCR.TXT
	USFC.CAL				
	USFC.PRN		1		
	USFCPR.TXT			1	USFCPR.TXT
LIVESTOCK PRODS.	USLA.TXT	1			USLAR.TXT
	USLA.CAL				USLAC.TXT
	USLA.PRN	1			
	USLS.TXT	1			USLSR.TXT
	USLS.CAL				USLSC.TXT
	USLS.PRN	1			
	USLN.TXT	1			USLNR.TXT
	USLN.CAL				USLNC.TXT
	USLN.PRN	1			
	USQL.TXT	1			USQLR.TXT
	USQL.CAL				USQLC.TXT
	USQL.PRN	1			
	USQO.TXT	1			USQOR.TXT
	USQO.CAL				USQOC.TXT
	USQO.PRN	1			
	USLNUM.TXT			1	USLNUM.TXT
DAIRY PRODUCTS	USDPDM.TXT			1	USDPDM.TXT
	USDP.TXT	1			USDPR.TXT
	USDP.CAL				USDPC.TXT
	USDP.PRN	1			
FEED DEMAND	USQFPI.TXT			1	USQFPI.TXT
	USQF.TXT	1			USQFR.TXT
	USQF.CAL				USQFC.TXT
	USQF.PRN	1			
INDUSTRIAL DEM.	USQIEQ.TXT			1	USQIEQ.TXT
FOOD+N.FEED DEM.	USQD.TXT	1			USQDR.TXT
	USQD.CAL				USQDC.TXT
	USQD.PRN	1			
STOCK DEMAND	USSK.TXT	1			USSKR.TXT
	USSK.CAL				
	USSK.PRN	1			
	USSKCF.TXT			1	USSKCF.TXT
NET TRADE	USQT.TXT		1		USQTR.TXT
	USQT.CAL				
	USQT.PRN		1		
TOTAL SUPPLY	USTS.TXT		1		USTSR.TXT
	USTS.CAL				
	USTS.PRN		1		
TOTAL DEMAND	USTD.TXT		1		USTDR.TXT
	USTD.CAL				
	USTD.PRN		1		
PRICE RATIO	USPR.TXT		1		USPRR.TXT
	USPR.CAL				
	USPR.PRN		1		

EQUATION TYPE	INPUT DATA FILE	---PROGRAM TO BE USED---- EQWRITE	EQDUPLIC	WORDSTAR	OUTPUT FILE
PRICE ESTIMATE	USPE.TXT		1		USPER.TXT
	USPE.CAL				
	USPE.PRN		1		
PRICE CONDITION	USPC.TXT		1		USPCR.TXT
	USPC.CAL				
	USPC.PRN		1		
DEMAND PRICE	USPD.TXT		1		USPDR.TXT
	USPD.CAL				
	USPD.PRN		1		
STANDARD REGION MODEL (RWGOL)					
SUPPLY	RWQS.TXT	1			RWQSR.TXT
	RWQS.CAL				RWQSC.TXT
	RWQS.PRN	1			
DEMAND	RWQD.TXT	1			RWQDR.TXT
	RWQD.CAL				RWQDC.TXT
	RWQD.PRN	1			
NET TRADE	RWQT.TXT		1		RWQTR.TXT
	RWQT.CAL				
	RWQT.PRN		1		
PRICE RATIO	RWPR.TXT		1		RWPRR.TXT
	RWPR.CAL				
	RWPR.PRN		1		
PRICE ESTIMATE	RWPE.TXT		1		RWPER.TXT
	RWPE.CAL				
	RWPE.PRN		1		
INTERNAL PRICE	RWPI.TXT		1		RWPIR.TXT
	RWPI.CAL				
	RWPI.PRN		1		
WORLD MARKET CLEARING MECHANISM (WDGOL)					
WORLD TRADE	*WDT.TXT				WDTR.TXT
	WD.CAL				
	*WDT.PRN				
ABS. VAL. WORLD TRADE	*WDA.TXT				WDAR.TXT
	WD.CAL				
	*WDA.PRN				
PRICE RATIO	WDPR.TXT		1		WDPRR.TXT
	WDPR.CAL				
	WDPR.PRN		1		
PRICE ESTIMATE	WDPE.TXT		1		WDPER.TXT
	WDPE.CAL				
	WDPE.PRN		1		
WORLD TRADE PRICE	WDPT.TXT		1		WDPTR.TXT
	WDPT.CAL				
	WDPT.PRN		1		

* REQUIRES PROGRAM 'CTYLINK' TO LINK COUNTRY DATA

II-C.1 Templates and Input Matrices for the Creation of GOL Equations for a Country Model

II-C.1.a. Domestic Margin

DOMESTIC MARGIN USMD.TXT
USMD.CAL
USMD.PRN

USMD.TXT
USMD$$
*(USPNG'X/USPD$$'N)**USMD$$##'C
*(USPNG'X(-1)/USPD$$'N(-1))**USMD$$##'C
*USPD$$'N

	A	B	C
1	USMD.PRN	PC	PL
2			
3	BF	.37	0
4	PK	.27	0
5	ML	0	0
6	DM	0	0
7	PM	0	0
8	PE	0	0
9	WH	0	0
10	CN	1	0
11	CG	1	0
12	RI	0	0
13	SB	0	0
14	OS	0	0
15	SM	1	0
16	SO	.02	0
17	OM	1	.4
18	OO	0	0
19	DB	0	0
20	DC	0	0
21	DO	0	0

USMD.PRN	PC	PL
BF	.37	0
PK	.27	0
ML	0	0
DM	0	0
PM	0	0
PE	0	0
WH	0	0
CN	1	0
CG	1	0
RI	0	0
SB	0	0
OS	0	0
SM	1	0
SO	.02	0
OM	1	.4
OO	0	0
DB	0	0
DC	0	0
DO	0	0

II-C.1.b. Trade Margin

TRADE MARGIN USMT.TXT
USMT.CAL
USMT.PRN

USMT.TXT
USMT$$
*(USPNG'X/USPD$$'N)**USMT$$##'C
*(USPNG'X(-1)/USPD$$'N(-1))**USMT$$##'C
*USPD$$'N

	A	B	C
1	USMT.PRN	PC	PL
2			
3	BF	.72	.09
4	PK	.3	0
5	ML	0	0
6	PM	.32	0
7	PE	0	0
8	WH	0	0
9	CN	1	0
10	CG	1	0
11	RI	1	0
12	SB	0	0
13	OS	0	0
14	SM	1	0
15	SO	1	0
16	OM	1	0
17	OO	0	0
18	DB	0	0
19	DC	0	0
20	DO	0	0

USMT.PRN	PC	PL
BF	.72	.09
PK	.3	0
ML	0	0
PM	.32	0
PE	0	0
WH	0	0
CN	1	0
CG	1	0
RI	1	0
SB	0	0
OS	0	0
SM	1	0
SO	1	0
OM	1	0
OO	0	0
DB	0	0
DC	0	0
DO	0	0

II-C.1.c. Supply Price

SUPPLY PRICE

USPS.TXT
USPS.CAL
USPS.PRN

```
USPS.TXT
USPS$$
'DEF ==
 ABSV'F(USPD$$'N
-USTC$$'POL
-USMD$$'N
-USTP$$'POL
)
```

	A	B	C	D	E	F
1	USPS.PRN	uspd$$	-ustc$$	-usmd$$	-ustp$$	)
2						
3	BF	1	0	1	0	1
4	PK	1	0	1	0	1
5	ML	1	0	1	0	1
6	DM	1	0	1	0	1
7	PM	1	0	1	0	1
8	PE	1	0	1	0	1
9	WH	1	0	1	0	1
10	CN	1	0	1	0	1
11	CG	1	0	1	0	1
12	RI	1	0	1	0	1
13	SB	1	0	1	0	1
14	OS	1	0	1	0	1
15	SM	1	0	1	0	1
16	SO	1	0	1	0	1
17	OM	1	0	1	0	1
18	OO	1	0	1	0	1
19	DB	1	0	1	0	1
20	DC	1	0	1	0	1
21	DO	1	0	1	0	1

USPS.PRN	uspd$$	-ustc$$	-usmd$$	-ustp$$	)
BF	1	0	1	0	1
PK	1	0	1	0	1
ML	1	0	1	0	1
DM	1	0	1	0	1
PM	1	0	1	0	1
PE	1	0	1	0	1
WH	1	0	1	0	1
CN	1	0	1	0	1
CG	1	0	1	0	1
RI	1	0	1	0	1
SB	1	0	1	0	1
OS	1	0	1	0	1
SM	1	0	1	0	1
SO	1	0	1	0	1
OM	1	0	1	0	1
OO	1	0	1	0	1
DB	1	0	1	0	1
DC	1	0	1	0	1
DO	1	0	1	0	1

II-C.1.d. Crop Area

```
CROP AREA          USTTRL.TXT
                   USARTT.TXT
                   USAR.TXT
                   USAR.CAL
                   USAR.PRN
```

```
USTTRL.TXT
BEGINNING OF FILE
-------------------------------------------------
USTTRL: USTTRL'DEF == (
USPSWH'DEF*USYDWH'N*USARWH'N
+USPSCN'DEF*USYDCN'N*USARCN'N
+USPSCG'DEF*USYDCG'N*USARCG'N
+USPSRI'DEF*USYDRI'N*USARRI'N
+USPSSB'DEF*USYDSB'N*USARSB'N
+USPSOS'DEF*USYDOS'N*USAROS'N
*100)/(USICP'X*(
 USARWH'N
+USARCN'N
+USARCG'N
+USARRI'N
+USARSB'N
+USAROS'N
)),
-------------------------------------------------
DO USTTRL = (
 USPSWH*USYDWH*USARWH
+USPSCN*USYDCN*USARCN
+USPSCG*USYDCG*USARCG
+USPSRI*USYDRI*USARRI
+USPSSB*USYDSB*USARSB
+USPSOS*USYDOS*USAROS
*100)/(USICP*(
 USARWH
+USARCN
+USARCG
+USARRI
+USARSB
+USAROS
));
-------------------------------------------------
END OF FILE
```

```
USARTT.TXT
BEGINNING OF FILE
-------------------------------------------------------
USARTT: USARTT'N=USARTTI'C*USTTRL'DEF(-1)**USARTTRL'C
*(1+USARTTGR'C)**TIME'X,
-------------------------------------------------------
USARTTRL 0.12,
USARTTTR 0.00341,
-------------------------------------------------------
DOCORE INTERCPT=USARTT/(USTTRL(-1)**USARTTRL'C
*(1+USARTTGR'C)**TIME'X);
DO USARTTI'C=MEAN(INTERCPT);
-------------------------------------------------------
END OF FILE
```

```
USAR.TXT
USAR$$
*(USPS##'DEF(-1)*USYD##'N(-1)/USICP'X(-1))**USAR$$##'C
*USARTT'N
```

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	CROP AREA ELASTICITIES			UNITED STATES				(with symmetry - top right related to bottom left by relative area)					
2													
3	USAR.CAL	WH	CN	CG	RI	SB	OS	Row SUM			AREA 80	AR SHARE	
4													
5	WH	.15	-.06	-.03	-.01	-.04	-.01	.00			28692	26.41	
6	CN	-.06	.09	-.01	-.01	-.01	-.01	.00			29583	27.23	
7	CG	-.07	-.02	.15	-.01	-.04	-.01	.00			11842	10.90	
8	RI	-.11	-.11	-.04	.28	-.01	-.01	.00			1333	1.23	
9	SB	-.04	-.01	-.02	.00	.08	-.01	.00			28557	26.29	
10	OS	-.03	-.03	-.01	.00	-.03	.12	.00			8630	7.94	
11													
12	Wtd. sum	.00	.00	.00	.00	.00	.00				108637	100.00	

USAR.CAL	WH	CN	CG	RI	SB	OS
WH	.15	-.06	-.03	-.01	-.04	-.01
CN	-.06	.09	-.01	-.01	-.01	-.01
CG	-.07	-.02	.15	-.01	-.04	-.01
RI	-.11	-.11	-.04	.28	-.01	-.01
SB	-.04	-.01	-.02	.00	.08	-.01
OS	-.03	-.03	-.01	.00	-.03	.12

CROP YIELD

USYD.TXT
USYD.CAL
USYD.PRN

USYD.TXT
USYD$$
*(USPS$$'DEF/USPIN'X)*SUSYD$$$$'C
*USAR$$'N**USYD$$##'C
*(1+USYD$$##'C)**TIME'X
*USWIN'X

II-C.1.e. Crop Yield

	A	B	C	D
1	USYD.PRN		AR	GR
2				
3	WH	.007	-.344	.01964
4	CN	.114	-.145	.02461
5	CG	.133	-.089	.01716
6	RI	.053	-.05	.00732
7	SB	.007	-.001	.00977
8	OS	.01	-.001	.010999

USYD.PRN		AR	GR
WH	.007	-.344	.01964
CN	.114	-.145	.02461
CG	.133	-.089	.01716
RI	.053	-.05	.00732
SB	.007	-.001	.00977
OS	.01	-.001	.010999

CROP SUPPLY

USPS.TXT
USPS.CAL
USPS.PRN

USPS.TXT
USPS$$
'DEF ==
 ABSV'F(USPD$$'N
-USTC$$'POL
-USMD$$'N
-USTP$$'POL
)

II-C.1.f. Crop Supply

	A	B	C	D	E	F
1	USPS.PRN	uspd$$	-ustc$$	-usmd$$	-ustp$$	)
2						
3	BF	1	0	1	0	1
4	PK	1	0	1	0	1
5	ML	1	0	1	0	1
6	DM	1	0	1	0	1
7	PM	1	0	1	0	1
8	PE	1	0	1	0	1
9	WH	1	0	1	0	1
10	CN	1	0	1	0	1
11	CG	1	0	1	0	1
12	RI	1	0	1	0	1
13	SB	1	0	1	0	1
14	OS	1	0	1	0	1
15	SM	1	0	1	0	1
16	SO	1	0	1	0	1
17	OM	1	0	1	0	1
18	OO	1	0	1	0	1
19	DB	1	0	1	0	1
20	DC	1	0	1	0	1
21	DO	1	0	1	0	1

USPS.PRN	uspd$$	-ustc$$	-usmd$$	-ustp$$	)
BF	1	0	1	0	1
PK	1	0	1	0	1
ML	1	0	1	0	1
DM	1	0	1	0	1
PM	1	0	1	0	1
PE	1	0	1	0	1
WH	1	0	1	0	1
CN	1	0	1	0	1
CG	1	0	1	0	1
RI	1	0	1	0	1
SB	1	0	1	0	1
OS	1	0	1	0	1
SM	1	0	1	0	1
SO	1	0	1	0	1
OM	1	0	1	0	1
OO	1	0	1	0	1
DB	1	0	1	0	1
DC	1	0	1	0	1
DO	1	0	1	0	1

II-C.1.g. Oilseed Products

OILSEED PRODUCTS USOILS.TXT

```
USOILS.TXT
BEGINNING OF FILE
-----------------------------------------------------------
USSBPM: USSBPM'DEF
(USOSSBSM'P*USPSSM'DEF+USOSSBSO'P*USPSSO'DEF)/USPDSB'N,
USOSPM: USOSPM'DEF ==
(USOSOSOM'P*USPSOM'DEF+USOSOSOO'P*USPSOO'DEF)/USPDOS'N,
-----------------------------------------------------------
DO USSBPM=(USOSSBSM'P*USPSSM+USOSSBSO'P*USPSSO)/USPDSB;
DO USOSPM=(USOSOSOM'P*USPSOM+USOSOSOO'P*USPSOO)/USPDOS;
-----------------------------------------------------------
USOCSB: USOCSB'N = USOCSBI'C
*USSBPM'DEF**USOCSBPN'C*(1+USOCSBGR'C)**TIME'X,
USOCOS: USOCOS'N = USOCOSI'C
*USOSPM'DEF**USOCOSPM'C*(1+USOCOSGR'C)**TIME'X,
-----------------------------------------------------------
USOSSBSM    0.8 ,
USOSSBSO    0.18,
USOSOSOM    0.5 ,
USOSOSOO    0.18,
USOCSBPM    0.01,
USOCSBTR    0.0498,
USOCOSPM    0.00,
USOCOSTR    0.0346,
-----------------------------------------------------------
DOCORE INTERCPT = USOCSB
/(USSBPM**USOCSBPM'C*(1+USOCSBGR'C)**TIME'X);
DO USOCSBI'C=MEAN(INTERCPT);
DOCORE INTERCPT = USOCOS
/(USOSPM**USOCOSPM'C*(1+USOCOSGR'C)**TIME'X);
DO USOCOSI'C=MEAN(INTERCPT);
-----------------------------------------------------------
USOSSM: USOSSM'DEF == USOSSBSM'P*USOCSB'N,
USOSSO: USOSSO'DEF == USOSSBSO'P*USOCSB'N,
USOSOM: USOSOM'DEF == USOSOSOM'P*USOCOS'N,
USOSOO: USOSOO'DEF == USOSOSOO'P*USOCOS'N,
-----------------------------------------------------------
DO USOSSM = USOSSBSM'P*USOCSB;
DO USOSSO = USOSSBSO'P*USOCSB;
DO USOSOM = USOSOSOM'P*USOCOS;
DO USOSOO = USOSOSOO'P*USOCOS;
-----------------------------------------------------------
END OF FILE
```

II-C.1.h. Feed Costs

FEED COSTS

USFC.TXT
USFC.CAL
USFC.PRN
USFCPR.TXT

USFC.TXT

```
USFC$$
'DEF ==
 USFC$$WH'P*USPDWH'N
+USFC$$CN'P*USPDCN'N
+USFC$$CG'P*USPDCG'N
+USFC$$SM'P*USPDSM'N
+USFC$$OM'P*USPDOM'N
```

	A	B	C	D	E	F
1	USFC.PRN	wheat	corn o. c. g.		soymeal o. meals	
2						
3	BF	1	1	1	1	1
4	PK	1	1	1	1	1
5	ML	1	1	1	1	1
6	DM	1	1	1	1	1
7	PM	1	1	1	1	1
8	PE	1	1	1	1	1

USFC.PRN	wheat	corn o. c. g.		soymeal o. meals	
BF	1	1	1	1	1
PK	1	1	1	1	1
ML	1	1	1	1	1
DM	1	1	1	1	1
PM	1	1	1	1	1
PE	1	1	1	1	1

USFCPR.TXT
BEGINNING OF FILE

```
USFCBFWH     0.01  ,
USFCPKWH     0.01  ,
USFCMLWH     0.01  ,
USFCDMWH     0.01  ,
USFCPMWH     0.01  ,
USFCPEWH     0.01  ,
USFCBFCN     0.60  ,
USFCPKCN     0.87  ,
USFCMLCN     0.60  ,
USFCDMCN     0.56  ,
USFCPMCN     0.60  ,
USFCPECN     0.49  ,
USFCBFCG     0.24  ,
USFCPKCG     0.09  ,
USFCMLCG     0.24  ,
USFCDMCG     0.35  ,
USFCPMCG     0.07  ,
USFCPECG     0.25  ,
USFCBFSM     0.15  ,
USFCPKSM     0.03  ,
USFCMLSM     0.15  ,
USFCDMSM     0.08  ,
USFCPMSM     0.32  ,
USFCPESM     0.25  ,
USFCBFOM     0.001 ,
USFCPKOM     0.001 ,
USFCMLOM     0.001 ,
USFCDMOM     0.001 ,
USFCPMOM     0.001 ,
USFCPEOM     0.001 ,
```

END OF FILE

LIVESTOCK PRODS. USLA.TXT
USLA.CAL
USLA.PRN
USLS.TXT
USLS.CAL
USLS.PRN
USLN.TXT
USLN.CAL
USLN.PRN
USQL.TXT
USQL.CAL
USQL.PRN
USQO.TXT
USQO.CAL
USQO.PRN
USLNUM.TXT

II-C.1.i. Livestock Products

USLA.TXT
USLA$$
*(USPS$$'DEF/USFC$$'DEF)**USLA$$##'C
*(USPS$$'DEF(-1)/USFC$$'DEF(-1))**USLA$$##'C
*USLN$$'N

	A	B	C
1	USLA.PRN	PC	PL
2			
3	BF	-.096	.067
4	PK	-.213	.676
5	ML	-.189	.081

USLA.PRN	PC	PL
BF	-.096	.067
PK	-.213	.676
ML	-.189	.081

USLS.TXT
USLS$$
*(USPS$$'DEF/USFC$$'DEF)**USLS$$##'C
*(USPS$$'DEF(-1)/USFC$$'DEF(-1))**USLS$$##'C
*USLN$$'N

	A	B	C
1	USLS.PRN	PC	PL
2			
3	BF	.142	.001
4	PK	-.298	.324
5	ML	-.234	.0387

USLS.PRN	PC	PL
BF	.142	.001
PK	-.298	.324
ML	-.234	.0387

USLN.TXT
USLN$$
*(USPS$$'DEF/USFC$$'DEF)**USLN$$##'C
*(USPS$$'DEF(-1)/USFC$$'DEF(-1))**USLN$$##'C
*USLN$$'N(-1)**USLN$$##'C

	A	B	C	D
1	USLN.PRN	PC	PL	LG
2				
3	DM	-.036	.013	.964
4	PE	.0001	.0001	.0001

USLN.PRN	PC	PL	LG
DM	-.036	.013	.964
PE	.0001	.0001	.0001

```
USQL.TXT
USQS$$
*(USPS$$'DEF/USFC$$'DEF)**USQS$$##'C
*(USPS$$'DEF(-1)/USFC$$'DEF(-1))**USQS$$##'C
*(1+USQS$$##'C)**TIME'X
*USLS$$'N
```

```
  |   A    ||    B    ||    C    ||    D    |
1| USQL.PRN        PC         PL         GR        USQL.PRN        PC         PL         GR
2|
3|        BF      .118      .0001      .0001              BF      .118      .0001      .0001
4|        PK      .015      .0001      .0001              PK      .015      .0001      .0001
5|        ML     -.021       .007     .00756              ML     -.021       .007     .00756
```

```
USQO.TXT
USQS$$
*(USPS$$'DEF/USFC$$'DEF)**USQS$$##'C
*(USPS$$'DEF(-1)/USFC$$'DEF(-1))**USQS$$##'C
*(1+USQS$$##'C)**TIME'X
*USLN$$'N
```

```
  |   A    ||    B    ||    C    ||    D    |
1| USQO.PRN        PC         PL         GR        USQO.PRN        PC         PL         GR
2|
3|        DM      .015      .0001     .00005              DM      .015      .0001     .00005
4|        PM      .018      .0001     .00061              PM      .018      .0001     .00061
5|        PE    -.0133      -.005     .00529              PE    -.0133      -.005     .00529
```

```
USLNUM.TXT
BEGINNING OF FILE
-------------------------------------------------------------
USLNBF: USLNBF'N = USLNBF'N(-1)+USLABF(-1)-USLSBF'N(-1),
USLNPK: USLNPK'N = USLNPK'N(-1)+USLAPK(-1)-USLSPK'N(-1),
USLNML: USLNML'N = USLNML'N(-1)+USLAML(-1)-USLSML'N(-1),
END OF FILE
```

II-C.1.j. Dairy Products

```
DAIRY PRODUCTS      USDPDM.TXT
                    USDP.TXT
                    USDP.CAL
                    USDP.PRN
```

```
USDPDM.TXT
BEGINNING OF FILE
-------------------------------------------------------------------------------
USOMDM: USOMDM'DEF == USOSDM'N-USODDM'N ,
-------------------------------------------------------------------------------
DO USOMDM = USOSDM - USODDM ;
END OF FILE
```

```
TYPE B:USDP.TXT
USDP.TXT
USOS$$
*(USPS##'DEF/USPSDM'DEF)**USOS$$##'C
*USOMDM'DEF
```

	A	B	C	D	E	F	G	H	I
1	Dairy Product Supply Elasticities (with symmetry)								
2									
3	USDP.PRN	DB	DC	DO		PDDM		Dairyp80	Dairpsh
4									
5	DB	.32	-.10	.04		-.26		1790.00	47.78
6	DC	-.12	.37	-.05		-.20		1434.67	38.29
7	DO	.14	-.14	.20		-.20		522.00	13.93
8									
9	WTD SUM	.12	.07	.03				3746.67	100.00

USDP.PRN	DB	DC	DO
DB	.32	-.10	.04
DC	-.12	.37	-.05
DO	.14	-.14	.20

FEED DEMAND USQFPI.TXT
USQF.TXT
USQF.CAL
USQF.PRN

USQFPI.TXT
BEGINNING OF FILE

```
-----------------------------------------------------
USLPI: USLPI'DEF == USLPWTBF'P*USPSBF'DEF
+USLPWTPK'P*USPSPK'DEF+USLPWTML'P*USPSML'DEF
+USLPWTDM'P*USPSDM'DEF+USLPWTPM'P*USPSPM'DEF
+USLPWTPE'P*USPSPE'DEF,
USGCAU: USGCAU'DEF == USGCAUBF'P*USLNBF'N
+USGCAUPK'P*USLNPK'N+USGCAUML'P*USLNML'N
+USGCAUDM'P*USLNDM'N+USGCAUPM'P*USQSPM'N
+USGCAUPE'P*USLNPE'N,
-----------------------------------------------------
USLPWTBF  0.65,
USLPWTPK  0.08,
USLPWTML  0.01,
USLPWTDM  0.06,
USLPWTPM  0.12,
USLPWTPE  0.08,
USGCAUBF  1.02,
USGCAUPK  1.28,
USGCAUML  0.0 ,
USGCAUDM  0.46,
USGCAUPM  0.01,
USGCAUPE  0.0 ,
-----------------------------------------------------
DO USLPI=USLPWTBF'P*USPSBF+USLPWTPK'P*USPSPK'N
+USLPWTML'P*USPSML+USLPWTDM'P*USPSDM+USLPWTPM'P*USPSPM
+USLPWTPE'P*USPSPE;
DO USGCAU=USGCAUBF'P*USLNBF+USGCAUPK'P*USLNPK
+USGCAUML'P*USLNML+USGCAUDM'P*USLNDM+USGCAUPM'P*USQSPM
+USGCAUPE'P*USLNPE;
END OF FILE
```

USQF.TXT

```
USQF$$
*(USPD##'N/USLPI'DEF)**USQF$$##'C
*USGCAU'DEF
```

	A	B	C	D	E	F	G	H	I	J
1	Feed Demand Elasticities				United States					
2										
3	USQF.PRN	WH	CN	CG	SM	OM		Row sum	Feed $0	Feedshr
4										
5	WH	-1.00	.60	.16	.21	.03		.00	2.04	1.41
6	CN	.01	-.10	.06	.02	.01		.00	104.15	71.86
7	CG	.02	.33	-.45	.08	.04		.01	19.22	13.26
8	SM	.02	.12	.09	-.26	.05		.02	17.45	12.04
9	OM	.03	.50	.37	.42	-1.29		.03	2.08	1.44
10										
11	Wtd. sum	.00	.00	.00	.00	.00			144.94	100.00

USQF.PRN	WH	CN	CG	SM	OM
WH	-1.00	.60	.16	.21	.03
CN	.01	-.10	.06	.02	.01
CG	.02	.33	-.45	.08	.04
SM	.02	.12	.09	-.26	.05
OM	.03	.50	.37	.42	-1.29

II-C.1.1. Industrial Demand

```
INDUSTRIAL DEM.    USQIEQ.TXT

USQIEQ.TXT
BEGINNING OF FILE
-----------------------------------------------------------------
USQICN: USQICN'N = USQICNI'C*(USPDCN'N/USPNG'X)**USQICNCN'C
*(1+USQICNGR'C)**TIME'X,
-----------------------------------------------------------------
USQICNCN  0.0  ,
USQICNTR  0.047,
USQICNI 0, USQICNCN 0, USQICNTR,
-----------------------------------------------------------------
DOCORE INTERCPT=USQICN/((USPDCN/USPNG)**USQICNCN'C
*(1+USQICNGR'C)**TIME);
DO USQICNI'C=MEAN(INTERCPT);
END OF FILE
```

II-C.1.m. Food and Non-Feed Demand

```
FOOD+N.FEED DEM.   USQD.TXT
                   USQD.CAL
                   USQD.PRN


USQD.TXT
USQD$$
*(USPD##'N/USPNG'X)**USQD$$##'C
*(USINC'X/(USPNG'X*USPOP'X))**USQD$$##'C
*USPOP'X
```

Food (and other) Demand Elasticities — United States — (with symmetry- applying demand theory)

USQD/DIF	BF	PK	ML	DM	PM	PE	WH	CN	CB	RI	SB	OS	SM
BF	-.64	.08	.00	.00	.07	.01	.00	.00	.00	.00	.00	.00	.00
PK	.20	-.41	.06	.00	.05	.00	.00	.00	.00	.00	.00	.00	.00
ML	-.01	1.55	-2.65	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
DM	.00	.00	.00	-.35	.00	.01	.00	.00	.00	.00	.00	.00	.00
PM	.61	.18	.00	.00	-.77	.00	.00	.00	.00	.00	.00	.00	.00
PE	.16	.00	.00	.04	.00	-.32	.00	.00	.00	.00	.00	.00	.00
WH	.01	.00	.00	.00	.00	.00	-.30	.00	.00	.05	.00	.00	.00
CN	-.01	.00	.00	.00	.00	.00	.00	-.10	.03	.00	.00	.00	.00
CB	.00	.00	.00	.00	.00	.00	.00	.07	-.03	.00	.00	.00	.00
RI	.01	.00	.00	.00	.00	.00	.16	.00	.00	-.32	.00	.00	.00
SB	.01	.00	.00	.00	.00	.00	.00	.00	.00	.00	-.01	.00	.00
OS	.01	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	-.01	.00
SM	.01	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	-.01
SO	-.03	-.01	.00	-.01	.00	.00	.00	.00	.00	.00	.00	.00	.00
OM	.01	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
OO	-.02	-.01	.00	-.01	.00	.00	.00	.00	.00	.00	.00	.00	.00
DB	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
DC	.06	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
DO	.01	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00

USQD/DIF	SO	OM	OO	DB	DC	DO	IN	Rowsum ElwrtPNB	Price 80	Quant 80	Value 80	Valshr80
BF	.00	.00	.00	.00	.01	.00	.29	-.18	5238.00	10866.00	56916.11	.03
PK	.00	.00	.00	.00	.00	.00	.13	.03	3075.00	7640.00	23493.00	.01
ML	.00	.00	.00	.00	.00	.00	.57	-.54	5571.00	163.00	908.07	.00
DM	.00	.00	.00	.00	.00	.00	.20	-.14	527.00	26359.00	13891.19	.01
PM	.00	.00	.00	.00	.00	.00	.18	.20	1032.00	6386.00	6590.35	.00
PE	.00	.00	.00	.00	.00	.00	.06	-.06	937.00	4020.00	3766.74	.00
WH	.00	.00	.00	.00	.00	.00	.08	-.16	159.00	19595.00	3115.61	.00
CN	.00	.00	.00	.00	.00	.00	.46	.37	111.00	16486.00	1829.95	.00
CB	.00	.00	.00	.00	.00	.00	.30	.33	148.00	5347.00	791.36	.00
RI	.00	.00	.00	.00	.00	.00	.06	-.09	562.00	1781.00	1000.92	.00
SB	.00	.00	.00	.00	.00	.00	.01	.01	282.00	2305.00	650.01	.00
OS	.00	.00	.00	.00	.00	.00	.01	.01	542.00	2103.00	1139.83	.00
SM	.00	.00	.00	.00	.00	.00	.01	.01	359.00	1.00	.36	.00
SO	-.15	.00	.11	.68	.00	.00	1.10	1.68	518.00	4073.00	2109.81	.00
OM	.00	-.01	.00	.00	.00	.00	.01	.02	344.00	211.00	72.58	.00
OO	.41	.00	-3.12	1.25	.00	.00	1.00	-.52	620.00	922.00	571.64	.00
DB	.75	.00	.38	-.65	.00	.00	.32	.79	4140.00	460.00	1904.40	.00
DC	.00	.00	.00	.00	-.46	.00	.25	-.15	5058.00	1793.00	9068.99	.01
DO	.00	.00	.00	.00	.00	-.05	.01	-.02	3469.00	354.00	1228.03	.00
									Othprd->	>>>>>>>>	>>>>>>>>	.92
												1.00
											1672.80	

USQD/DIF	BF	PK	ML	DM	PM	PE	WH	CN	CB	RI	SB	OS	SM
BF	-.64	.08	.00	.00	.07	.01	.00	.00	.00	.00	.00	.00	.00
PK	.20	-.41	.06	.00	.05	.00	.00	.00	.00	.00	.00	.00	.00
ML	-.01	1.55	-2.65	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
DM	.00	.00	.00	-.35	.00	.01	.00	.00	.00	.00	.00	.00	.00
PM	.61	.18	.00	.00	-.77	.00	.00	.00	.00	.00	.00	.00	.00
PE	.16	.00	.00	.04	.00	-.32	.00	.00	.00	.00	.00	.00	.00
WH	.01	.00	.00	.00	.00	.00	-.30	.00	.00	.05	.00	.00	.00
CN	-.01	.00	.00	.00	.00	.00	.00	-.10	.03	.00	.00	.00	.00
CB	.00	.00	.00	.00	.00	.00	.00	.07	-.03	.00	.00	.00	.00
RI	.01	.00	.00	.00	.00	.00	.16	.00	.00	-.32	.00	.00	.00
SB	.01	.00	.00	.00	.00	.00	.00	.00	.00	.00	-.01	.00	.00
OS	.01	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	-.01	.00
SM	.01	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	-.01
SO	-.03	-.01	.00	-.01	.00	.00	.00	.00	.00	.00	.00	.00	.00
OM	.01	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
OO	-.02	-.01	.00	-.01	.00	.00	.00	.00	.00	.00	.00	.00	.00
DB	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
DC	.06	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
DO	.01	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00

SO	OM	OO	DB	DC	DO	IN
.00	.00	.00	.00	.01	.00	.29
.00	.00	.00	.00	.00	.00	.13
.00	.00	.00	.00	.00	.00	.57
.00	.00	.00	.00	.00	.00	.20
.00	.00	.00	.00	.00	.00	.18
.00	.00	.00	.00	.00	.00	.06
.00	.00	.00	.00	.00	.00	.08
.00	.00	.00	.00	.00	.00	.46
.00	.00	.00	.00	.00	.00	.30
.00	.00	.00	.00	.00	.00	.06
.00	.00	.00	.00	.00	.00	.01
.00	.00	.00	.00	.00	.00	.01
.00	.00	.00	.00	.00	.00	.01
-.15	.00	.11	.68	.00	.00	1.10
.00	-.01	.00	.00	.00	.00	.01
.41	.00	-3.12	1.25	.00	.00	1.00
.75	.00	.38	-.65	.00	.00	.32
.00	.00	.00	.00	-.46	.00	.25
.00	.00	.00	.00	.00	-.05	.01

STOCK DEMAND

USSK.TXT
USSK.CAL
USSK.PRN
USSKCF.TXT

II-C.1.n. Stock Demand

USSK.TXT

```
USSK$$
*(USPD$$'N/USPNG'X)**USSK$$$$'C
*(USPR$$'POL /USPS$$'DEF)**USSK$$PR'C
*(USQD$$'N
+USQF$$'N
+USQC$$'N
+USQI$$'N
+USQS$$'N)
+USQS$$'DEF)
```

	A	B	C	D	E	F	G	H	I
1	USSK.PRN	own pr	parity	qd	qf	qc	qi	qsn	qsdef
2									
3	BF	0	0	1	0	0	0	1	0
4	PK	-.15	0	1	0	0	0	1	0
5	ML	0	0	1	0	0	0	1	0
6	PM	0	0	1	0	0	0	1	0
7	PE	0	0	1	0	0	0	1	0
8	WH	-.91	0	1	1	0	0	0	1
9	CN	-1.45	0	1	1	0	1	0	1
10	CG	-1.35	0	1	1	0	0	0	1
11	RI	-.24	0	1	0	0	0	0	1
12	SB	0	0	1	0	1	0	0	1
13	OS	0	0	1	0	1	0	0	1
14	SM	-1	0	1	1	0	0	0	1
15	SO	0	0	1	0	0	0	0	1
16	OM	-1	0	1	1	0	0	0	1
17	OO	0	0	1	0	0	0	0	1
18	DB	0	1	1	0	0	0	0	0
19	DC	0	1	1	0	0	0	1	0
20	DO	0	1	1	0	0	0	1	0

USSK.PRN	own pr	parity	qd	qf	qc	qi	qsn	qsdef
BF	0	0	1	0	0	0	1	0
PK	-.15	0	1	0	0	0	1	0
ML	0	0	1	0	0	0	1	0
PM	0	0	1	0	0	0	1	0
PE	0	0	1	0	0	0	1	0
WH	-.91	0	1	1	0	0	0	1
CN	-1.45	0	1	1	0	1	0	1
CG	-1.35	0	1	1	0	0	0	1
RI	-.24	0	1	0	0	0	0	1
SB	0	0	1	0	1	0	0	1
OS	0	0	1	0	1	0	0	1
SM	-1	0	1	1	0	0	0	1
SO	0	0	1	0	0	0	0	1
OM	-1	0	1	1	0	0	0	1
OO	0	0	1	0	0	0	0	1
DB	0	1	1	0	0	0	0	0
DC	0	1	1	0	0	0	1	0
DO	0	1	1	0	0	0	1	0

```
USSKCF.TXT
BEGINNING OF FILE
-----------------------------------------------------------------
USSKBFI'C 0, USSKPKI'C 0, USSKMLI'C 0, USSKPMI'C 0, USSKPEI'C 0, USSKWHI'C 0,
USSKCNI'C 0, USSKCGI'C 0, USSKRII'C 0, USSKSBI'C 0, USSKOSI'C 0, USSKSMI'C 0,
USSKSOI'C 0, USSKOMI'C 0, USSKOOI'C 0, USSKDBI'C 0, USSKDCI'C 0, USSKDOI'C 0,
USSKPKPK     0, USSKWHWH     0, USSKCNCN     0, USSKCGCG     0, USSKRIRI    0,
USSKSMSM     0, USSKOMOM     0, USSKDBPR 0, USSKDCPR 0, USSKDOPR 0,
-----------------------------------------------------------------
USSKPKPK   -0.15 ,
USSKWHWH   -0.91 ,
USSKCNCN   -1.45 ,
USSKCGCG   -1.35 ,
USSKRIRI   -0.24 ,
USSKSMSM   -1.00 ,
USSKOMOM   -1.00 ,
USSKDBPR    1.00 ,
USSKDCPR    1.00 ,
USSKDOPR    1.00 ,
END OF FILE
```

NET TRADE

USQT.TXT
USQT.CAL
USQT.PRN

```
USQT.TXT
USQT$$
'N =
USQS$$
'N
'DEF
-USQC$$'N
-USQI$$'N
-USQF$$'N
-USQD$$'N-(USSK$$'N-USSK$$'N(-1))
```

II-C.1.o. Net Trade

	A	B	C	D	E	F	G	H
1	USQT.PRN	qs	qs'n	qs'def	qc	qi	qf	qd sk
2								
3	BF	1	1	0	0	0	0	1
4	PK	1	1	0	0	0	0	1
5	ML	1	1	0	0	0	0	1
6	PM	1	1	0	0	0	0	1
7	PE	1	1	0	0	0	0	1
8	WH	1	1	1	0	0	1	1
9	CN	1	0	1	0	1	1	1
10	CG	1	0	1	0	0	1	1
11	RI	1	0	1	0	0	0	1
12	SB	1	0	1	1	0	0	1
13	OS	1	0	1	1	0	0	1
14	SM	1	0	1	0	0	1	1
15	SO	1	0	1	0	0	0	1
16	OM	1	0	1	0	0	1	1
17	OO	1	0	1	0	0	0	1
18	DB	1	1	0	0	0	0	1
19	DC	1	1	0	0	0	0	1
20	DO	1	1	0	0	0	0	1

USQT.PRN	qs	qs'n	qs'def	qc	qi	qf	qd sk
BF	1	1	0	0	0	0	1
PK	1	1	0	0	0	0	1
ML	1	1	0	0	0	0	1
PM	1	1	0	0	0	0	1
PE	1	1	0	0	0	0	1
WH	1	1	1	0	0	1	1
CN	1	0	1	0	1	1	1
CG	1	0	1	0	0	1	1
RI	1	0	1	0	0	0	1
SB	1	0	1	1	0	0	1
OS	1	0	1	1	0	0	1
SM	1	0	1	0	0	1	1
SO	1	0	1	0	0	0	1
OM	1	0	1	0	0	1	1
OO	1	0	1	0	0	0	1
DB	1	1	0	0	0	0	1
DC	1	1	0	0	0	0	1
DO	1	1	0	0	0	0	1

TOTAL SUPPLY

II-C.1.p. Total Supply

```
USTS.TXT
USTS$$
'DEF ==
USQS$$
'DEF
'N
```

	A	B	C	D
1	USTS.PRN	qs	'def	'n
2				
3	BF	1	0	1
4	PK	1	0	1
5	ML	1	0	1
6	PM	1	0	1
7	PE	1	0	1
8	WH	1	1	0
9	CN	1	1	0
10	CG	1	1	0
11	RI	1	1	0
12	SB	1	1	0
13	OS	1	1	0
14	SM	1	1	0
15	SO	1	1	0
16	OM	1	1	0
17	OO	1	1	0
18	DB	1	0	1
19	DC	1	0	1
20	DO	1	0	1

USTS.PRN	qs	'def	'n
BF	1	0	1
PK	1	0	1
ML	1	0	1
PM	1	0	1
PE	1	0	1
WH	1	1	0
CN	1	1	0
CG	1	1	0
RI	1	1	0
SB	1	1	0
OS	1	1	0
SM	1	1	0
SO	1	1	0
OM	1	1	0
OO	1	1	0
DB	1	0	1
DC	1	0	1
DO	1	0	1

TOTAL DEMAND

II-C.1.q. Total Demand

```
USTD.TXT
USTD$$
'DEF ==
USQD$$'N
+USQF$$'N
+USQC$$'N
+USQI$$'N
+(USSK$$'N-USSK$$'N(-1))
```

	A	B	C	D	E	F
1	USTD.PRN	qd	+qf	+qc	+qi	+dsk
2						
3	BF	1	0	0	0	1
4	PK	1	0	0	0	1
5	ML	1	0	0	0	1
6	PM	1	0	0	0	1
7	PE	1	0	0	0	1
8	WH	1	1	0	0	1
9	CN	1	1	0	1	1
10	CG	1	1	0	0	1
11	RI	1	0	0	0	1
12	SB	1	0	1	0	1
13	OS	1	0	1	0	1
14	SM	1	1	0	0	1
15	SO	1	0	0	0	1
16	OM	1	1	0	0	1
17	OO	1	0	0	0	1
18	DB	1	0	0	0	1
19	DC	1	0	0	0	1
20	DO	1	0	0	0	1

USTD.PRN	qd	+qf	+qc	+qi	+dsk
BF	1	0	0	0	1
PK	1	0	0	0	1
ML	1	0	0	0	1
PM	1	0	0	0	1
PE	1	0	0	0	1
WH	1	1	0	0	1
CN	1	1	0	1	1
CG	1	1	0	0	1
RI	1	0	0	0	1
SB	1	0	1	0	1
OS	1	0	1	0	1
SM	1	1	0	0	1
SO	1	0	0	0	1
OM	1	1	0	0	1
OO	1	0	0	0	1
DB	1	0	0	0	1
DC	1	0	0	0	1
DO	1	0	0	0	1

USPR.TXT PRICE RATIO

USPR$$ II-C.1.r. Price Ratio

```
'DEF ==
USCP$$'P*(USOT$$'N-(
IF (USOT$$'N GE USEO$$'POL) THEN USEO$$'POL ELSE
IF (USOT$$'N LE -USMO$$'POL) THEN -USMO$$'POL ELSE
0))/(USTS$$'DEF+USTD$$"DEF)
```

	A	B	C	D	E
1	USPR.PRN	cp*qt-	eq	-mqO/(ts+td)	
2					
3	BF	1	1	1	1
4	PK	1	1	1	1
5	ML	1	1	1	1
6	PM	1	1	1	1
7	PE	1	1	1	1
8	WH	1	1	1	1
9	CN	1	1	1	1
10	CG	1	1	1	1
11	RI	1	1	1	1
12	SB	1	1	1	1
13	OS	1	1	1	1
14	SM	1	1	1	1
15	SO	1	1	1	1
16	OM	1	1	1	1
17	OO	1	1	1	1
18	DB	1	1	1	1
19	DC	1	1	1	1
20	DO	1	1	1	1

USPR.PRN	cp*qt-	eq	-mqO/(ts+td)	
BF	1	1	1	1
PK	1	1	1	1
ML	1	1	1	1
PM	1	1	1	1
PE	1	1	1	1
WH	1	1	1	1
CN	1	1	1	1
CG	1	1	1	1
RI	1	1	1	1
SB	1	1	1	1
OS	1	1	1	1
SM	1	1	1	1
SO	1	1	1	1
OM	1	1	1	1
OO	1	1	1	1
DB	1	1	1	1
DC	1	1	1	1
DO	1	1	1	1

USPE.TXT PRICE ESTIMATE II-C.l.s. Price Estimate

```
USPE$$
'DEF ==
USPD$$'N*(1-(
 IF USPR$$'DEF GT USCL$$'P THEN USCL$$'P ELSE
IF USPR$$'DEF LT -USCL$$'P THEN -USCL$$'P ELSE
USPR$$'DEF))
```

	A	B	C	D	E
1	USPE.PRN	pd*1-	cl	-cl	pr
2					
3	BF	1	1	1	1
4	PK	1	1	1	1
5	ML	1	1	1	1
6	PM	1	1	1	1
7	PE	1	1	1	1
8	WH	1	1	1	1
9	CN	1	1	1	1
10	CG	1	1	1	1
11	RI	1	1	1	1
12	SB	1	1	1	1
13	OS	1	1	1	1
14	SM	1	1	1	1
15	SO	1	1	1	1
16	OM	1	1	1	1
17	OO	1	1	1	1
18	DB	1	1	1	1
19	DC	1	1	1	1
20	DO	1	1	1	1

USPE.PRN	pd*1-	cl	-cl	pr
BF	1	1	1	1
PK	1	1	1	1
ML	1	1	1	1
PM	1	1	1	1
PE	1	1	1	1
WH	1	1	1	1
CN	1	1	1	1
CG	1	1	1	1
RI	1	1	1	1
SB	1	1	1	1
OS	1	1	1	1
SM	1	1	1	1
SO	1	1	1	1
OM	1	1	1	1
OO	1	1	1	1
DB	1	1	1	1
DC	1	1	1	1
DO	1	1	1	1

USPC.TXT PRICE CONDITION

```
USPC$$
'DEF ==
1*(((USPE$$'DEF LT (USPT$$'N+USMT$$'N
+USTM$$'POL
+USTC$$'POL
)) AND (USPE$$'DEF GT (USPT$$'N-USMT$$'N+USMD$$'N
-USTE$$'POL
+USTC$$'POL
))) OR
(USQT$$'N GE USEQ$$'POL)
OR
(USQT$$'N LE -USMQ$$'POL)
)
```

II-C.l.t. Price Condition

	A	B	C	D	E	F	G	H	I	J	K	L	
1	USPC.PRN	pe<pt	+tm	+tc	pe>pt	-te	+tc	or	qt>eq	or	qt>-mq		)
2													
3	BF	1	1	1	1	1	1	1	1	1	1	1	1
4	PK	1	1	1	1	1	1	1	1	1	1	1	1
5	ML	1	1	1	1	1	1	1	1	1	1	1	1
6	PM	1	1	1	1	1	1	1	1	1	1	1	1
7	PE	1	1	1	1	1	1	1	1	1	1	1	1
8	WH	1	1	1	1	1	1	1	1	1	1	1	1
9	CN	1	1	1	1	1	1	1	1	1	1	1	1
10	CG	1	1	1	1	1	1	1	1	1	1	1	1
11	RI	1	1	1	1	1	1	1	1	1	1	1	1
12	SB	1	1	1	1	1	1	1	1	1	1	1	1
13	OS	1	1	1	1	1	1	1	1	1	1	1	1
14	SM	1	1	1	1	1	1	1	1	1	1	1	1
15	SO	1	1	1	1	1	1	1	1	1	1	1	1
16	OM	1	1	1	1	1	1	1	1	1	1	1	1
17	OO	1	1	1	1	1	1	1	1	1	1	1	1
18	DB	1	1	1	1	1	1	1	1	1	1	1	1
19	DC	1	1	1	1	1	1	1	1	1	1	1	1
20	DO	1	1	1	1	1	1	1	1	1	1	1	1

USPC.PRN	pe<pt	+tm	+tc	pe>pt	-te	+tc	or	qt>eq	or	qt>-mq	)
BF	1	1	1	1	1	1	1	1	1	1	1
PK	1	1	1	1	1	1	1	1	1	1	1
ML	1	1	1	1	1	1	1	1	1	1	1
PM	1	1	1	1	1	1	1	1	1	1	1
PE	1	1	1	1	1	1	1	1	1	1	1
WH	1	1	1	1	1	1	1	1	1	1	1
CN	1	1	1	1	1	1	1	1	1	1	1
CG	1	1	1	1	1	1	1	1	1	1	1
RI	1	1	1	1	1	1	1	1	1	1	1
SB	1	1	1	1	1	1	1	1	1	1	1
OS	1	1	1	1	1	1	1	1	1	1	1
SM	1	1	1	1	1	1	1	1	1	1	1
SO	1	1	1	1	1	1	1	1	1	1	1
OM	1	1	1	1	1	1	1	1	1	1	1
OO	1	1	1	1	1	1	1	1	1	1	1
DB	1	1	1	1	1	1	1	1	1	1	1
DC	1	1	1	1	1	1	1	1	1	1	1
DO	1	1	1	1	1	1	1	1	1	1	1

II-C.l.u. Demand price

DEMAND PRICE

```
USPD.TXT
USPD$$
'N =
IF (USPC$$'DEF EQ 1) THEN ABSV'F(USPE$$'DEF)
ELSE
IF (USQT$$'N LT 0) THEN
ABSV'F(USPT$$'N+USMT$$'N
+USTM$$'POL
+USTC$$'POL
)
ELSE
ABSV'F(USPT$$'N-USMT$$'N+USMD$$'N
-USTE$$'POL
+USTC$$'POL
)
```

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	USPD.PRN	pc+1	else	qt<0	pt	+tm	+tc	)	else	pt	-te	+tc	)
2													
3	BF	1	1	1	1	1	1	1	1	1	1	1	1
4	PK	1	1	1	1	1	1	1	1	1	1	1	1
5	ML	1	1	1	1	1	1	1	1	1	1	1	1
6	PM	1	1	1	1	1	1	1	1	1	1	1	1
7	PE	1	1	1	1	1	1	1	1	1	1	1	1
8	WH	1	1	1	1	1	1	1	1	1	1	1	1
9	CN	1	1	1	1	1	1	1	1	1	1	1	1
10	CG	1	1	1	1	1	1	1	1	1	1	1	1
11	RI	1	1	1	1	1	1	1	1	1	1	1	1
12	SB	1	1	1	1	1	1	1	1	1	1	1	1
13	OS	1	1	1	1	1	1	1	1	1	1	1	1
14	SM	1	1	1	1	1	1	1	1	1	1	1	1
15	SO	1	1	1	1	1	1	1	1	1	1	1	1
16	OM	1	1	1	1	1	1	1	1	1	1	1	1
17	OO	1	1	1	1	1	1	1	1	1	1	1	1
18	DB	1	1	1	1	1	1	1	1	1	1	1	1
19	DC	1	1	1	1	1	1	1	1	1	1	1	1
20	DO	1	1	1	1	1	1	1	1	1	1	1	1
21													
22													

TYPE B:USPD.TXT

USPD.PRN	pc+1	else	qt<0	pt	+tm	+tc	)	else	pt	-te	+tc	)
BF	1	1	1	1	1	1	1	1	1	1	1	1
PK	1	1	1	1	1	1	1	1	1	1	1	1
ML	1	1	1	1	1	1	1	1	1	1	1	1
PM	1	1	1	1	1	1	1	1	1	1	1	1
PE	1	1	1	1	1	1	1	1	1	1	1	1
WH	1	1	1	1	1	1	1	1	1	1	1	1
CN	1	1	1	1	1	1	1	1	1	1	1	1
CG	1	1	1	1	1	1	1	1	1	1	1	1
RI	1	1	1	1	1	1	1	1	1	1	1	1
SB	1	1	1	1	1	1	1	1	1	1	1	1
OS	1	1	1	1	1	1	1	1	1	1	1	1
SM	1	1	1	1	1	1	1	1	1	1	1	1
SO	1	1	1	1	1	1	1	1	1	1	1	1
OM	1	1	1	1	1	1	1	1	1	1	1	1
OO	1	1	1	1	1	1	1	1	1	1	1	1
DB	1	1	1	1	1	1	1	1	1	1	1	1
DC	1	1	1	1	1	1	1	1	1	1	1	1
DO	1	1	1	1	1	1	1	1	1	1	1	1

II-C.2. Templates and Input Matrices for the Creation of GOL Equations for a a Regional Model

II-C.2.a. Supply

```
SUPPLY          RWQS.TXT                RWQS.TXT
                RWQS.CAL                RWQS$$
                RWQS.PRN                *(RWPI$$'N/RWPNG'X)**RWQS$$PC'C
                                        *(RWPI$$'N(-1)/RWPNG'X(-1))**RWQS$$PL'C
                                        *(1+RWQS$$TR'C)**TIME'X
                                        *RWQS$$'N(-1)**RWQS$$LG'C
                                        *RWQS$$SF'X
```

	A	B	C	D	E
1	RWQS.PRN	PC	PL	GR	LG
2					
3	BF	.01	0	0	.934
4	PK	.154	-.065	.048	0
5	ML	0	0	.009	0
6	PM	0	0	.0745	0
7	PE	0	.097	.034	0
8	WH	.0085	0	0	.913
9	CN	0	.006	.04	0
10	CG	0	0	.022	0
11	RI	0	0	.0304	0
12	SB	0	.395	.0946	0
13	OS	0	.114	.0516	0
14	SM	.048	.112	.111	0
15	SO	.106	.053	.122	0
16	OM	.295	.164	.0736	0
17	OO	.027	.091	.056	0
18	DB	.026	0	0	.772
19	DC	.223	0	0	.808
20	DO	.111	0	0	.852

RWQS.PRN	PC	PL	GR	LG
BF	.01	0	0	.934
PK	.154	-.065	.048	0
ML	0	0	.009	0
PM	0	0	.0745	0
PE	0	.097	.034	0
WH	.0085	0	0	.913
CN	0	.006	.04	0
CG	0	0	.022	0
RI	0	0	.0304	0
SB	0	.395	.0946	0
OS	0	.114	.0516	0
SM	.048	.112	.111	0
SO	.106	.053	.122	0
OM	.295	.164	.0736	0
OO	.027	.091	.056	0
DB	.026	0	0	.772
DC	.223	0	0	.808
DO	.111	0	0	.852

II-C.2.b. Demand

DEMAND

RWQD.TXT
RWQD.CAL
RWQD.PRN

```
RWQD.TXT
RWQD$$
*(RWPI$$'N/RWPNG'X)**RWQD$$$$'C
*(RWINC'X/(RWPNG'X*RWPOP'X))**RWQD$$##'C
*RWPOP'X
```

	A	B	C
1	RWQD.PRN		IN
2			
3	BF	-.17	.36
4	PK	-.09	.34
5	ML	-.24	-.38
6	PM	-.56	.87
7	PE	-.05	.22
8	WH	-.03	.78
9	CN	-.32	.02
10	CG	-.35	-1.19
11	RI	-.01	.12
12	SB	-.02	1.25
13	OS	-.01	.01
14	SM	-.99	1.86
15	SO	-.91	1.09
16	OM	-.11	-.26
17	OO	-.01	.52
18	DB	0	.26
19	DC	-.49	.17
20	DO	-.01	.12

RWQD.PRN		IN
BF	-.17	.36
PK	-.09	.34
ML	-.24	-.38
PM	-.56	.87
PE	-.05	.22
WH	-.03	.78
CN	-.32	.02
CG	-.35	-1.19
RI	-.01	.12
SB	-.02	1.25
OS	-.01	.01
SM	-.99	1.86
SO	-.91	1.09
OM	-.11	-.26
OO	-.01	.52
DB	0	.26
DC	-.49	.17
DO	-.01	.12

NET TRADE

RWQT.TXT
RWQT.CAL
RWQT.PRN

```
RWQT.TXT
RWQT$$
'N =
RWQS$$'N
-RWQD$$'N
```

II-C.2.c. Net Trade

	A	B	C
1	RWQT.PRN	qs	-qd
2			
3	BF	1	1
4	PK	1	1
5	ML	1	1
6	PM	1	1
7	PE	1	1
8	WH	1	1
9	CN	1	1
10	CG	1	1
11	RI	1	1
12	SB	1	1
13	OS	1	1
14	SM	1	1
15	SO	1	1
16	OM	1	1
17	OO	1	1
18	DB	1	1
19	DC	1	1
20	DO	1	1

RWQT.PRN	qs	-qd
BF	1	1
PK	1	1
ML	1	1
PM	1	1
PE	1	1
WH	1	1
CN	1	1
CG	1	1
RI	1	1
SB	1	1
OS	1	1
SM	1	1
SO	1	1
OM	1	1
OO	1	1
DB	1	1
DC	1	1
DO	1	1

II-C.2.d. Price Ratio

PRICE RATIO	RWPR.TXT
	RWPR.CAL
	RWPR.PRN
		RWPR.TXT
		RWPR$$
		'DEF ==
		RWCP$$'P*
		((RWOT$$'N-(
		IF RWOT$$'N GE RWEQ$$'POL THEN
		RWEQ$$'N
		ELSE
		-RWMQ$$'POL
		))/(RWOS$$'N+RWOD$$'N))

	A	B	C	D	E	F	G	H
1	RWPR.PRN	cp*	qt-if	qt>=eq	eq	else	-mq	/(qs+qd)
2								
3	BF	1	1	1	1	1	1	1
4	PK	1	1	1	1	1	1	1
5	ML	1	1	1	1	1	1	1
6	PM	1	1	1	1	1	1	1
7	PE	1	1	1	1	1	1	1
8	WH	1	1	1	1	1	1	1
9	CN	1	1	1	1	1	1	1
10	CG	1	1	1	1	1	1	1
11	RI	1	1	1	1	1	1	1
12	SB	1	1	1	1	1	1	1
13	OS	1	1	1	1	1	1	1
14	SM	1	1	1	1	1	1	1
15	SO	1	1	1	1	1	1	1
16	OM	1	1	1	1	1	1	1
17	OO	1	1	1	1	1	1	1
18	DB	1	1	1	1	1	1	1
19	DC	1	1	1	1	1	1	1
20	DO	1	1	1	1	1	1	1

RWPR.PRN	cp*	qt-if	qt>=eq	eq	else	-mq	/(qs+qd)
BF	1	1	1	1	1	1	1
PK	1	1	1	1	1	1	1
ML	1	1	1	1	1	1	1
PM	1	1	1	1	1	1	1
PE	1	1	1	1	1	1	1
WH	1	1	1	1	1	1	1
CN	1	1	1	1	1	1	1
CG	1	1	1	1	1	1	1
RI	1	1	1	1	1	1	1
SB	1	1	1	1	1	1	1
OS	1	1	1	1	1	1	1
SM	1	1	1	1	1	1	1
SO	1	1	1	1	1	1	1
OM	1	1	1	1	1	1	1
OO	1	1	1	1	1	1	1
DB	1	1	1	1	1	1	1
DC	1	1	1	1	1	1	1
DO	1	1	1	1	1	1	1

PRICE ESTIMATE RWPE.TXT
RWPE.CAL
RWPE.PRN

RWPE.TXT

```
RWPE$$
'DEF ==
 IF ((RWQT$$'N LT RWEQ$$'POL) AND (RWQT$$'N GT -RWMQ$$'POL)) THEN
 WDPT$$'X
 ELSE RWPI$$'N*(1+(RWCP$$'P/(
 RWQS$$'N*RWQS$$PC
-RWQD$$'N*RWQD$$$$
))*(RWQT$$'N+(
 IF RWQT$$'N GE RWEQ$$'POL THEN
 -RWEQ$$'POL
 ELSE
 RWMQ$$'POL
)))
```

II-C.2.e. Price Estimate

	A	B	C	D	E	F	G	H	I	J	K	L
1	RWPE.PRN	if()then	wdpt	else pi	qses	-qded	*qt	if gt eq	-eq	else	mw	)))
2												
3	BF	1	1	1	1	1	1	1	1	1	1	1
4	PK	1	1	1	1	1	1	1	1	1	1	1
5	ML	1	1	1	1	1	1	1	1	1	1	1
6	PM	1	1	1	1	1	1	1	1	1	1	1
7	PE	1	1	1	1	1	1	1	1	1	1	1
8	WH	1	1	1	1	1	1	1	1	1	1	1
9	CN	1	1	1	1	1	1	1	1	1	1	1
10	CG	1	1	1	1	1	1	1	1	1	1	1
11	RI	1	1	1	1	1	1	1	1	1	1	1
12	SB	1	1	1	1	1	1	1	1	1	1	1
13	OS	1	1	1	1	1	1	1	1	1	1	1
14	SM	1	1	1	1	1	1	1	1	1	1	1
15	SO	1	1	1	1	1	1	1	1	1	1	1
16	OM	1	1	1	1	1	1	1	1	1	1	1
17	OO	1	1	1	1	1	1	1	1	1	1	1
18	DB	1	1	1	1	1	1	1	1	1	1	1
19	DC	1	1	1	1	1	1	1	1	1	1	1
20	DO	1	1	1	1	1	1	1	1	1	1	1

RWPE.PRN	if()then	wdpt	else pi	qses	-qded	*qt if qt eq	-eq	else	mw	)))
BF	1	1	1	1	1	1	1	1	1	1
PK	1	1	1	1	1	1	1	1	1	1
ML	1	1	1	1	1	1	1	1	1	1
PM	1	1	1	1	1	1	1	1	1	1
PE	1	1	1	1	1	1	1	1	1	1
WH	1	1	1	1	1	1	1	1	1	1
CN	1	1	1	1	1	1	1	1	1	1
CG	1	1	1	1	1	1	1	1	1	1
RI	1	1	1	1	1	1	1	1	1	1
SB	1	1	1	1	1	1	1	1	1	1
OS	1	1	1	1	1	1	1	1	1	1
SM	1	1	1	1	1	1	1	1	1	1
SO	1	1	1	1	1	1	1	1	1	1
OM	1	1	1	1	1	1	1	1	1	1
OO	1	1	1	1	1	1	1	1	1	1
DB	1	1	1	1	1	1	1	1	1	1
DC	1	1	1	1	1	1	1	1	1	1
DO	1	1	1	1	1	1	1	1	1	1

INTERNAL PRICE RWPI.TXT
RWPI.CAL
RWPI.PRN

```
RWPI.TXT
RWPI$$
'N =
 RWPE$$'DEF
```

	A	B
1	RWPI.PRN	pe
2		
3	BF	1
4	PK	1
5	ML	1
6	PM	1
7	PE	1
8	WH	1
9	CN	1
10	CG	1
11	RI	1
12	SB	1
13	OS	1
14	SM	1
15	SO	1
16	OM	1
17	OO	1
18	DB	1
19	DC	1
20	DO	1

RWPI.PRN	pe
BF	1
PK	1
ML	1
PM	1
PE	1
WH	1
CN	1
CG	1
RI	1
SB	1
OS	1
SM	1
SO	1
OM	1
OO	1
DB	1
DC	1
DO	1

II-C.2.f. Internal Price

II-C.3. Templates and Input Matrices for the Creation of GOL Equations for the World Market Clearing Mechanism

II-C.3.a. World Trade

```
WORLD
TRADE
```

```
WDT.TXT
WD.CAL
WDT.PRN
```

```
WDT.TXT
WDOT$$
'DEF ==
 ##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+##OT$$
+RWOT$$
```

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	WDT	US	CN	EC	WE	JP	AZ	SF	EE	SV	CH	MX	CA	BZ	AR	VE	LA	AF	EG	ME	ND	OS	DO	TH	SA	EA	OA
2																											
3	BF	1	1	1	_	1	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_
4	PK	1	1	1	_	1	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_
5	ML	1	1	1	_	1	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_
6	PM	1	1	1	_	1	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_
7	PE	1	1	1	_	1	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_
8	WH	1	1	1	_	1	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_
9	CN	1	1	1	_	1	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_
10	CG	1	1	1	_	1	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_
11	RI	1	1	1	_	1	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_
12	SB	1	1	1	_	1	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_
13	OS	1	1	1	_	1	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_
14	SM	1	1	1	_	1	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_
15	SO	1	1	1	_	1	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_
16	OM	1	1	1	_	1	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_
17	OO	1	1	1	_	1	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_
18	DB	1	1	1	_	0	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_
19	DC	1	1	1	_	0	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_
20	DO	1	1	1	_	0	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_

WDT	US	CN	EC	WE	JP	AZ	SF	EE	SV	CH	MX	CA	BZ	AR	VE	LA	AF	EG	ME	ND	OS	DO	TH	SA	EA	OA	RW
BF	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
PK	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
ML	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
PM	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
PE	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
WH	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
CN	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
CG	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
RI	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
SB	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
OS	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
SM	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
SO	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
OM	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
OO	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
DB	1	1	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
DC	1	1	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
DO	1	1	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1

ABS. VAL.
WORLD
TRADE

WDA.TXT
WD.CAL
WDA.PRN

II-C.3.b. Absolute Value of World Trade

```
WDA.TXT
WDAT$$
'DEF ==
 ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(##QT$$)
+ABSV'F(RWQT$$)
```

WDA	US	CN	EC	WE	JP	AZ	SF	EE	SV	CH	MX	CA	BZ	AR	VE	LA	AF	EG	ME	ND	OS	DO	TH	SA	EA	OA	RW
BF	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
PK	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
ML	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
PM	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
PE	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
WH	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
CN	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
CG	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
RI	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
SB	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
OS	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
SM	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
SO	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
OM	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
OO	1	1	1	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
DB	1	1	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
DC	1	1	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1
DO	1	1	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1

PRICE RATIO

WDPR.TXT
WDPR.CAL
WDPR.PRN

```
WDPR.TXT
WDPR$$
'DEF ==
WDCP$$'P
*WDQT$$'DEF
/(1+WDAT$$'DEF)
```

II-C.3.c. Price Ratio

	A	B	C	D
1	WDPR.PRN	wdcp	wdqt	/wdat
2				
3	BF	1	1	1
4	PK	1	1	1
5	ML	1	1	1
6	PM	1	1	1
7	PE	1	1	1
8	WH	1	1	1
9	CN	1	1	1
10	CG	1	1	1
11	RI	1	1	1
12	SB	1	1	1
13	OS	1	1	1
14	SM	1	1	1
15	SO	1	1	1
16	OM	1	1	1
17	OO	1	1	1
18	DB	1	1	1
19	DC	1	1	1
20	DO	1	1	1

WDPR.PRN	wdcp	wdqt	/wdat
BF	1	1	1
PK	1	1	1
ML	1	1	1
PM	1	1	1
PE	1	1	1
WH	1	1	1
CN	1	1	1
CG	1	1	1
RI	1	1	1
SB	1	1	1
OS	1	1	1
SM	1	1	1
SO	1	1	1
OM	1	1	1
OO	1	1	1
DB	1	1	1
DC	1	1	1
DO	1	1	1

PRICE ESTIMATE

WDPE.TXT
WDPE.CAL
WDPE.PRN

```
WDPE.TXT
WDPE$$
'DEF ==
WDPT$$'N*(IF
(WDPR$$'DEF GT WDCL$$'P)
THEN
(1-WDCL$$'P)
ELSE
(IF
(WDPR$$'DEF LT -WDCL$$'P)
THEN
(1+WDCL$$'P)
ELSE
(1-WDPR$$'DEF)
))
```

II-C.3.d. Price Estimate

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	WDPE.PRN	wdpt*	if pr>cl	then	1-cl	else	(if	pr<cl	then	1+cl	else	1-pr	)
2													
3	BF	1	1	1	1	1	1	1	1	1	1	1	1
4	PK	1	1	1	1	1	1	1	1	1	1	1	1
5	ML	1	1	1	1	1	1	1	1	1	1	1	1
6	PM	1	1	1	1	1	1	1	1	1	1	1	1
7	PE	1	1	1	1	1	1	1	1	1	1	1	1
8	WH	1	1	1	1	1	1	1	1	1	1	1	1
9	CN	1	1	1	1	1	1	1	1	1	1	1	1
10	CG	1	1	1	1	1	1	1	1	1	1	1	1
11	RI	1	1	1	1	1	1	1	1	1	1	1	1
12	SB	1	1	1	1	1	1	1	1	1	1	1	1
13	OS	1	1	1	1	1	1	1	1	1	1	1	1
14	SM	1	1	1	1	1	1	1	1	1	1	1	1
15	SO	1	1	1	1	1	1	1	1	1	1	1	1
16	OM	1	1	1	1	1	1	1	1	1	1	1	1
17	OO	1	1	1	1	1	1	1	1	1	1	1	1
18	DB	1	1	1	1	1	1	1	1	1	1	1	1
19	DC	1	1	1	1	1	1	1	1	1	1	1	1
20	DO	1	1	1	1	1	1	1	1	1	1	1	1

WDPE.PRN	wdpt*	if pr>cl	then	1-cl	else	(if	pr<cl	then	1+cl	else	1-pr	)
BF	1	1	1	1	1	1	1	1	1	1	1	1
PK	1	1	1	1	1	1	1	1	1	1	1	1
ML	1	1	1	1	1	1	1	1	1	1	1	1
PM	1	1	1	1	1	1	1	1	1	1	1	1
PE	1	1	1	1	1	1	1	1	1	1	1	1
WH	1	1	1	1	1	1	1	1	1	1	1	1
CN	1	1	1	1	1	1	1	1	1	1	1	1
CG	1	1	1	1	1	1	1	1	1	1	1	1
RI	1	1	1	1	1	1	1	1	1	1	1	1
SB	1	1	1	1	1	1	1	1	1	1	1	1
OS	1	1	1	1	1	1	1	1	1	1	1	1
SM	1	1	1	1	1	1	1	1	1	1	1	1
SO	1	1	1	1	1	1	1	1	1	1	1	1
OM	1	1	1	1	1	1	1	1	1	1	1	1
OO	1	1	1	1	1	1	1	1	1	1	1	1
DB	1	1	1	1	1	1	1	1	1	1	1	1
DC	1	1	1	1	1	1	1	1	1	1	1	1
DO	1	1	1	1	1	1	1	1	1	1	1	1

II-C.3.e. World Trade Price Estimate

```
WORLD TRADE PRICE WDPT.TXT
                  WDPT.CAL
                  WDPT.PRN
```

```
WDPT.TXT
WDPT$$
'N =
WDPE$$'DEF
```

	A	B
1	WDPT.PRN	wdpe
2		
3	BF	1
4	PK	1
5	ML	1
6	PM	1
7	PE	1
8	WH	1
9	CN	1
10	CG	1
11	RI	1
12	SB	1
13	OS	1
14	SM	1
15	SO	1
16	OM	1
17	OO	1
18	DB	1
19	DC	1
20	DO	1

WDPT.PRN	wdpe
BF	1
PK	1
ML	1
PM	1
PE	1
WH	1
CN	1
CG	1
RI	1
SB	1
OS	1
SM	1
SO	1
OM	1
OO	1
DB	1
DC	1
DO	1

III. Concluding Comments

The programs referenced in this report allow a researcher a variety of ways to develop models in TROLL for GOL (or other purposes). Appropriate reference manuals must be consulted for details about statistical/simulation packages such as TROLL and SAS. Reference materials must also be consulted when using mainframe programs/systems such as TSO or micro computer programs/languages such as SUPERCALC and BASIC.

A researcher usually begins a modeling exercise by choosing a model specification and assembling appropriate data. To the extent that a specification can be standardized or can follow some of the GOL standard specifications, the programs in this report can greatly simplify the modeling task. For convenience, data can be entered into SUPERCALC on a micro computer and can be transformed to the TROLL data format by the program SCTROLL. TROLLFRM can be used to set up a TROLL file to contain models and data. Models or parts thereof can be cloned from existing U.S. versions via mainframe computer techniques presented in this report. Alternatively, the researcher can begin with the templates and input matrices in this report, modify them to suit his specification, and create groups of model equations for TROLL on the micro computer using EQWRITE and EQDUPLIC. As is true with the data, once TROLL statements/equations are created, they can be telecommunicated to the mainframe computer for entry into the user's TROLL file.

Various micro computer utility programs help the researcher with the preparation of data and model equations. Mainframe computer routines and TROLL 'MACRO' programs can help the researcher to simulate his model and display the simulation results.

IV. Appendix - Program to convert SUPERCALC file to MICROSTAT file

This program converts a SUPERCALC xx. PRN file to a direct access file which can be read by MICROSTAT (trademark). The SUPERCALC file must have variable names (no more than 20) at the top of data columns. See references manuals on MICROSTAT for details on using this package.

SCMICRO.BAS

```
10 REM PROGRAM CREATED BY VERNON OLEY RONINGEN - MAR. 24, 1983
20 REM
30 REM THIS PROGRAM REQUIRES AN INPUT SUPERCALC .PRN FILE AS FOLLOWS:
40 REM LINE 1 - VARIABLE NAMES OVER DATA COLUMNS (NO NAME MEANS COLUMN IS IGNORED
50 REM LINE 2 - BLANK LINE
60 REM LINE 3 - 1ST OBSERVATION OF ALL VARIABLES
70 REM LINE 4 - 2ND OBSERVATION OF ALL VARIABLES
80 REM LINE 4 - ETC.
90 REM          NOTE THAT A MICROSTAT ACCEPTS A MAXIMUM OF 20 VARIABLES AND THAT
100 REM          VARIABLES NAMES MUST BE 5 CHARS. OR LESS.  FILE NAMES MUST BE 7
110 REM
120 CLEAR 5000
130 GOSUB 1110
140 DIM D(25,50),V$(1,50)
150 PRINT" S C M I C R O   BY VOR":PRINT
160 PRINT"PROGRAM TO CONVERT A SUPERCALC ***.PRN DATA FILE TO A DIRECT ACCESS"
170 PRINT"FILE READABLE BY MICROSTAT (TRADEMARK)":PRINT
180 INPUT"ENTER 'NAME' OF SUPERCALC (NAME.PRN) FILE";N$
190 F$="B:"+N$+".PRN"
200 REM ---------READ SUPERCALC FILE
210 PRINT:PRINT"READING FILE ";F$:PRINT
220 OPEN"I",1,F$
230 I=0:J1=1
240 LINE INPUT #1,W$
250 J0=LEN(W$)/9
260 FOR J=1 TO J0
270 S$=MID$(W$,(J-1)*9+1,9)
280 GOSUB 1060
290 V$(1,J1+J-1)=S$
300 NEXT J
310 LINE INPUT #1,W$
320 IF EOF(1) THEN 420
330 LINE INPUT #1,W$
340 IF VAL(LEFT$(W$,9))=0 AND MID$(W$,9,1)<>"0" THEN 350 ELSE 370
350 I=0:J1=J0+1
360 GOTO 250
370 I=I+1
380 FOR J=1 TO J0
390 D(I,J1+J-1)=VAL(MID$(W$,(J-1)*9+1,9))
400 NEXT J
```

```
410 GOTO 320
420 CLOSE 1:PRINT:PRINT"FILE ";F$;" IS CLOSED":PRINT
430 J0=J1 +J0-1
440 REM ---------PRINT FILE READ IN
450 PRINT"THE SUPERCALC FILE READ IN IS:":PRINT
460 FOR J=1 TO J0
470 PRINT V$(1,J),
480 FOR I1=1 TO I
490 PRINT D(I1,J);
500 NEXT I1
510 PRINT
520 NEXT J
530 REM ---------COUNT VARAIBLES TO INCLUDE IN MICROSTAT FILE
540 M=0
550 FOR J=1 TO J0
560 IF LEN(V$(1,J))<>0 THEN M=M+1
570 NEXT J
580 IF M>20 THEN M=20
590 REM ----------CONVERT TO MICROSTAT EXAMPLE NOTATION
600 N=I
610 Z$=N$+"R"
620 Q5=4
630 C$=N$+" DATA"
640 O$="B:"+N$
650 DIM A$(J0):SP$=SPACE$(5)
660 OPEN"O",1,O$
670 REM ----------WRITE MICROSTAT HEADER FILE
680 PRINT:PRINT"WRITING MICROSTAT HEADER FILE ";O$:PRINT
690 PRINT #1,Q5;",";N;",";M;",";C$;",";
700 PRINT Q5;",";N;",";M;",";C$;",";
710 FOR J=1 TO J0
720 IF LEN(V$(1,J))=0 THEN 780
730 IF J>20 THEN 780
740 RSET SP$=V$(1,J)
750 A$(J)=SP$
760 PRINT #1,A$(J)
770 PRINT A$(J)
780 NEXT J
790 Z$="B:"+Z$
800 PRINT #1,Z$
810 CLOSE 1
820 PRINT Z$
830 PRINT:PRINT"FILE ";O$;" IS CLOSED":PRINT
840 REM ---------WRITE MICROSTAT DATA FILE
850 PRINT:PRINT"WRITING MICROSTAT DIRECT ACCESS DATA FILE.";Z$
860 PRINT:OPEN"R",#1,Z$,Q5:FIELD #1,Q5 AS T$
870 FOR J=1 TO N
880 K1=0
890 FOR K=1 TO J0
900 IF LEN(V$(1,K))=0 THEN 980
910 IF K>20 THEN 980
920 K1=K1+1
930 Y=D(J,K)
940 J1=K1+((J-1)*M)
950 PRINT Y;
960 LSET T$=MKS$(Y)
970 PUT #1,J1
980 NEXT K
990 NEXT J
1000 PRINT
1010 CLOSE 1
1020 PRINT:PRINT"FILE ";Z$;" IS CLOSED":PRINT
1030 PRINT"MICROSTAT FILE ";O$;" IS READY":PRINT
1040 PRINT " THE FILE HAS ";I;" OBSERVATIONS AND ";M;" VARIABLES":PRINT
1050 END
1060 REM----------SUBROUTINE TO REMOVE LEFT BLANKS FROM S$
1070 P0=INSTR(S$," ")
1080 IF P0=0 THEN RETURN
1090 S$=RIGHT$(S$,LEN(S$)-1)
1100 GOTO 1070
1110 REM ----------SUBROUTINE TO CLEAR SCREEN
1120 PRINT CHR$(27);CHR$(58):RETURN
```

Sample input SUPERCALC xx. PRN file

YEAR	ONE	TWO	THREE	FOUR	FIVE	SIX	SEVEN	EIGHT	NINE	TEN	ELEVEN	TWELVE	THIRTEEN	FOURTEEN	FIFTEEN
1961	1	1	5	100	2019	1	5	1	1.2386e8	159	52	-123456	1	0	2.5836e8
1962	2	2	4	111	1621	2	4	1	1.3265e8	753	123	-1234562	0	2	-1.594e8
1963	3	3	3	152	1532	3	3	0	1.2390e8	258	-352	-4.569e7	0	5	-1.596e8
1964	4	4	2	163	1235	4	2	0	1.2365e8	456	-200	-456951	1	3	3.5795e8
1965	5	5	1	141	1000	5	1	0	1.2346e8	987	-100	-2583571	1	1	0

Sample file when printed out using MICROSTAT

HEADER FOR: B:TEST
LABEL: TEST DATA
CASES: 5 VARIABLES: 16

	YEAR	ONE	TWO	THREE	FOUR	FIVE
1	1961.00	1.00	1.00	5.00	100.00	2019.00
2	1962.00	2.00	2.00	4.00	111.00	1621.00
3	1963.00	3.00	3.00	3.00	152.00	1532.00
4	1964.00	4.00	4.00	2.00	163.00	1235.00
5	1965.00	5.00	5.00	1.00	141.00	1000.00

	SIX	SEVEN	EIGHT	NINE	TEN	ELEVE
1	1.00	5.00	1.00	%123860000.00	159.00	52.00
2	2.00	4.00	1.00	%132650000.00	753.00	123.00
3	3.00	3.00	0.00	%123900000.00	258.00	-352.00
4	4.00	2.00	0.00	%123650000.00	456.00	-200.00
5	5.00	1.00	0.00	%123460000.00	987.00	-100.00

	TWELV	THIRT	FOURT	FIFTE
1	-123456.00	1.00	0.00	%258360000.00
2	%-1234560.00	0.00	2.00	%-1594000000.00
3	%-45690000.00	0.00	5.00	%-1596000000.00
4	-456951.00	1.00	3.00	%357950000.00
5	%-2583570.00	1.00	1.00	0.00